2011

A Thermoelectric generation subsystem model for heat recovery simulations

Andrew Freedman

A Thermoelectric Generation Subsystem Model for Heat Recovery Simulations

Andrew P. Freedman

A Thesis Submitted in Partial Fulfillment of the Requirements for a Master of Science Degree in Mechanical Engineering

Approved by:

Dr. Robert Stevens – Thesis Advisor _____
Department of Mechanical Engineering

Dr. Margaret Bailey – Professor _____
Department of Mechanical Engineering

Dr. Steven Weinstein – Professor _____
Department of Mechanical Engineering
Department Head of Chemical Engineering

Dr. Edward Hensel – Department Representative _____
Department Head of Mechanical Engineering

Date of Approval: 7/20/2011

Department of Mechanical Engineering
Rochester Institute of Technology
Rochester, New York 14623

July 2011

PERMISSION TO REPRODUCE THESIS

A Thermoelectric Generation Subsystem Model for Heat Recovery Simulations

I, ANDREW P. FREEDMAN, hereby grant permission to the Wallace Memorial Library of Rochester Institute of Technology to reproduce my thesis in the whole or part. Any reproduction will not be for commercial use or profit.

Date: _____

Signature: _____

July 2011

Abstract

Thermoelectric devices can convert thermal energy directly into electrical energy, or they can work in reverse and use electrical energy to create a temperature gradient for cooling or heating applications. The absence of moving parts, wide range of operating temperatures, scalability, and modular capabilities makes thermoelectrics attractive for energy generation applications. They have been considered for use with vehicle exhausts, co-generation, and other energy recovery from lost heat in thermodynamic cycles. Thermoelectric devices have relatively low efficiencies but there have been recent advances in thermoelectric materials potentially opening the door to more power applications. As material advancements continue and a wider range of power generation applications will be considered, module and system level modeling becomes critical for the design of the next generation of thermoelectric systems.

The overall focus of this thesis is to further develop the current thermoelectric power unit models available, validate the developed model, and implement the model into a simulation environment for feasibility and optimization studies. Current models found in the literature are

often based on very specific applications or are too general in nature to truly explore the optimization of a wide range of potential thermoelectric applications. The model developed in this work is highly customizable permitting the optimization of a large number of varying systems. Module mismatch and heat spreading in three dimensions are explored, modeled, and considered for incorporation into the new thermoelectric power unit model to allow for more accurate performance prediction.

Mismatch models are developed for thermoelectric modules electrically connected in series and in parallel. Predecessors in the research community believe power output to be hampered due to variation of module specific parameters. Developed models for performance prediction display the extent of the mismatch effect. Experimental validation of the model shows the high level of exactness in prediction.

Often, a one dimensional heat transfer assumption is made when analyzing system performance. This is known to cause discrepancies in the thermoelectric power unit model predictions. For thermal resistances with regards to the thermoelectric modules internal to the thermoelectric power unit, a heat spreading effect is observed. This effect requires an analytical method for quantifying the extent of the three dimensional resistance. The analytical model is described and applied to the thermoelectric power unit. Numerical comparison and experimentation are performed to verify the accuracy of the model. The model for heat spreading in three dimensions is incorporated into the new thermoelectric power unit model to allow for more accurate performance prediction.

The advanced thermoelectric power unit model is coded in the ThermoElectric Power System Simulator (TEPSS) environment for a project sponsored by New York State Energy Research and Development Authority (NYSERDA). The programmed module will serve as the key component of the software package that will predict the performance of the thermoelectric heat recovery unit used in common thermodynamic cycles. Experimental validation of the advanced model is performed. Data is compared to simulation results from the TEPSS *tepowerunit* component. The data fits are agreeable when the uncertainty of various system parameters is considered.

Acknowledgements

First and foremost, I'd like to thank Dr. Robert Stevens for allowing me to work on a Master's Thesis in his lab, providing superb guidance and direction throughout the duration of the thesis, and for going above and beyond an advisor by being a great friend.

Secondly, I'd like to thank all of the Rochester Institute of Technology Mechanical Engineering Department faculty and staff. The office staff, for always being available for guidance and support throughout the thesis process, the machine shop staff for their skill of the craft and assistance in times of need, and the faculty for the dissemination of their knowledge. Special thanks to Dr. Margaret Bailey and Dr. Steven Weinstein for serving on my committee and to Dr. Edward Hensel for being the department representative.

In addition, I'd like to thank New York State Energy Research and Development Authority (NYSERDA) for their funding of the project as well as Rochester Institute of Technology.

Finally, I'd like to thank my family and friends for their support throughout my thesis. Especially, my immediate family, thank you. My brother, Brad, for the "Victory" song which provided encouragement in a time of need. And most importantly, my parents, Paula and Gary, for their lifelong encouragement of success in my education as well as the specific inspiration in finishing my thesis when it seemed it was an impossible task. Thank you.

Contents

List of Figures

List of Tables

Nomenclature

Symbol	Parameter
A	Area [m]
a, b	Heat Spreading Sink Dimensions [m]
$Biot \cdot \tau$	Non-Dimensional Biot Number [-]
C_p or c_p	Specific Heat Capacity [J/(kg·K)]
C_1	Ratio of Seebeck Coefficients for Two Modules [-]
C_2	Ratio of Internal Electrical Resistances for Two Modules [-]
D	Diameter [m]
f	Friction Factor [-]
h	Convective Coefficient [W/(m^2·K)]
Hg	Hagen Number [-]
I	Current [A]
j	Colburn Factor [-]
L	Length [m]
$\dot{m}$	Mass Flow Rate [kg/s]
N	Number [-]
NTU	Number of Transfer Units [-]
Nu	Nusselt Number [-]
P	Perimeter [m]
P	Power [W]
P	Pressure [Pa]
p	Pitch [m]
Pr	Prandtl Number [-]
q or Q	Heat Rate [W]
R	Resistance - Electrical [Ω] or Thermal [K/W]
R''	Resistance Per Unit Area – Thermal [(m^2·K)/W] or Electrical [Ω·m^2]
Re	Reynolds Number [-]
S	Fin Spacing Distance [m]
T	Temperature [°C or K]
t	Thickness [m]
UA	UA-value [W/K]
V	Velocity [m/s]
V	Voltage [V]
V	Volume [m^3]
w	Width [m]
x, y, z	Directions [-]
ZT	Figure of Merit [-]

Symbol	Parameter
α or $\alpha_{p,n}$	Seebeck Coefficient [V/K]
ΔT	Temperature Difference [-]
Δx, Δy	Source Dimensions [m]
η	Efficiency [-]
λ or k	Thermal Conductivity [W/m·K]
μ	Viscosity [Pa·s]
ρ	Electrical Resistivity [Ω·m]
σ	Electrical Conductivity [S/m]
τ	Thomson Coefficient [μV/K]
Ψ	Non-Dimensional Thermal Resistance [-]

Subscript	Parameter
avg	Average
b	Base
bar	Aluminum Bar
C or *c*	Cool or Cold Side
CE	Cold Side Junction
cer	Ceramic
ch	Channel
char	Characteristic
columns	Columns
comb	Combined
con	Contact
e	Internal Electrical Resistance
eff	Effective
ent	Entrance
envir	Environmental
exp	Experimental
f	Fin
H or *h*	Hot or Warm Side
HE	Hot Side Junction
HX	Heat Exchanger
in or *i*	Inlet Position
ins	Insulation
L	Load
lam	Laminar

Subscript	Parameter
lm	Log Mean
LMTD	Log Mean Temperature Difference
long	Longitudinal
losses or *losses*	Heat Losses
max	Maximum
meas	Measured
mod	Module
model	Modeled
n	Negative Semiconductor
oc	Open Circuit
out or *o*	Outlet Position
p	Positive Semiconductor
p or *par*	Parallel
rows	Rows
s or *ser*	Series
s	Shunt
sc	Short Circuit
Sp	Spreading
surf	Surface
TEM	Thermoelectric Module
TE1 or *1*	Thermoelectric Module 1
TE2 or *2*	Thermoelectric Module 2
th	Thermal
Tot or *tot*	Total
trans	Transverse
turb	Turbulent
U	One Dimensional
zone or *z*	Zone
0	Overall
∞	Ambient Temperature

1 Introduction

Thermoelectric (TE) modules are devices that utilize semiconductor technology to either directly convert heat to electricity or use electricity to move heat. When used in waste heat recovery applications, thermoelectric generators could potentially benefit the thermodynamic systems and increase overall efficiency. The absence of moving parts, wide range of operating temperatures, scalability, and modular capabilities makes thermoelectric generators attractive for energy generation applications. They have been considered for use with vehicle exhausts, co-generation, and other energy recovery from lost heat in thermodynamic cycles. Until recently, thermoelectric devices have been only used by niche applications because of their low efficiencies and expensive materials. As manufacturing costs drop, better materials are developed, and module efficiencies increase, a powerful performance prediction tool is needed to predict and optimize potential power generation applications. Heat exchangers provide a reasonable device to harness thermoelectric generators and their recovery contributions. Interestingly, heat exchangers utilizing thermoelectric modules have been modeled but typically, only focus on improving module efficiency without accounting for production cost. Often, improving the performance of the heat exchanger itself is neglected or unrealistic assumptions are made to create inaccurate heat exchanger capability. An expansive model accounting for thermoelectric generator performance, heat exchanger performance, and associated costs with the ability to optimize design and implement the design into a thermodynamic cycle is developed for the Thermoelectric Power System Simulator (TEPSS) and described herein.

1.1 Background

Thermoelectric devices are solid state devices that generate electricity when a temperature gradient exists between each side of the thermoelectric device. Thermoelectric modules are driven by the basic principles of the Seebeck Effect and Peltier Effect. The Seebeck Effect governs the operation of each thermoelectric module (TEM) which contains two dissimilar thermoelectric materials. These are arranged electrically in series and thermally in parallel. The junctions of the different conductors are kept at different temperatures which cause an open circuit electromotive force (emf) to develop as follows:

$$V_{oc} = \alpha(T_H - T_C) \tag{1.1.1}$$

where α is the difference in Seebeck coefficient of the two leg materials and has the units of V/K, and T_H and T_C are the hot and cool side absolute temperatures both measured in Kelvin. Thomas Johann Seebeck, a German physicist, discovered this effect in the early 1800s. When a load is applied, electric current is driven by the temperature difference as heat conducts through the device. In the same time period, an independent discovery from Seebeck's research was made by French physicist, Jean Charles Athanase Peltier. The Peltier Effect is the reverse of the

Seebeck Effect; as electric current is applied to a semiconductor, heat is absorbed or released depending on direction of current and the relationship of the Seebeck coefficients. The heat absorption or dissipation may create a temperature gradient depending on thermal loading [1].

An important unit-less metric for evaluating the performance of thermoelectric materials is the thermoelectric figure of merit, *ZT*. It characterizes the effectiveness of a specific thermoelectric material in terms of its electrical and thermal material properties. The figure of merit is expressed as

$$ZT = \frac{\alpha^2 \sigma}{\lambda} T \qquad (1.1.2)$$

where α is the Seebeck coefficient, σ is the electrical conductivity, λ is the thermal conductivity, and *T* is the absolute average temperature [1, 2]. *ZT* for materials has remained below one for decades but in recent years, *ZT* of new materials has reached values greater than two. These increases in the figure of merit are a result of studies in nanostructures of thermoelectric materials. This research has the promise of increasing the figure of merit to even higher values. With values greater than 3, thermoelectrics provide feasible solutions in many applications [3].

Modules consist of a group of thermoelectric semiconductor pairs, called legs, connected electrically in series and thermally in parallel as is shown in Figure 1.1. The ends of the leg pairs are typically connected by a conductor. On top of these conductors an electrically insulating but thermally conductive material is attached to prevent electrical shorting of the leg pairs.

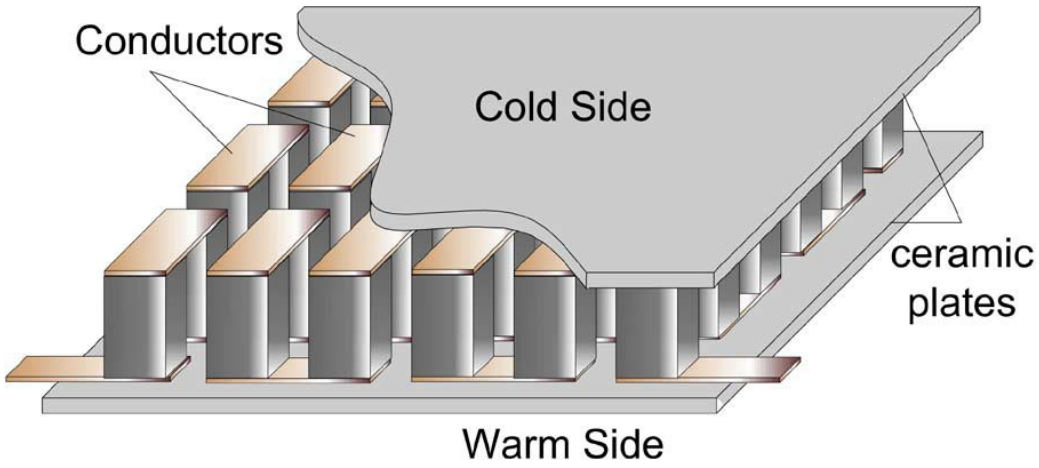


Figure 1.1: Typical one-stage 18-couple TEM with ceramic plates [4].

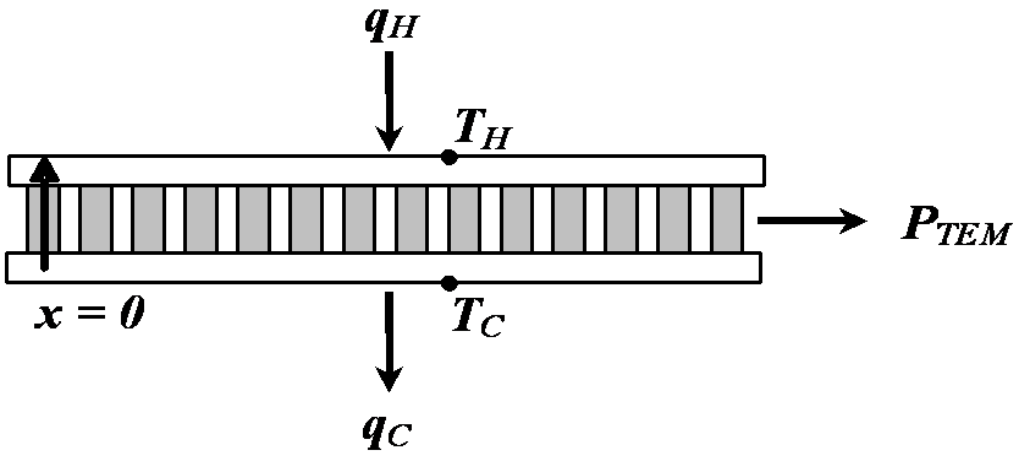


Figure 1.2: Schematic representation of thermoelectric generator [5].

Figure 1.2 represents the thermoelectric module in Figure 1.1 from a side view with thermal (q_H, q_C) and electrical (P_{TEM}) power represented. Module level Seebeck coefficient ($N\alpha$), module level thermal resistance (N/R_{th}), and module level electrical resistance (NR_e) are the three parameters that define the performance of a thermoelectric module where N represents the number of thermoelectric leg pairs. The heat flowing into the hot surface, q_H, and out of the cold surface, q_C, are given by

$$q_H = N\alpha I T_H + \frac{N}{R_{th}}(T_H - T_C) - \frac{1}{2}NR_e I^2 \quad (1.1.3)$$

$$q_C = N\alpha I T_C + \frac{N}{R_{th}}(T_H - T_C) + \frac{1}{2}NR_e I^2 \quad (1.1.4)$$

with I representing the current [6]. This model operates under the assumptions that there is 1-D thermal transport only, the Seebeck coefficient is constant, and contact resistance of the conducting strips and ceramic plates can be ignored. These equations are derived from the heat equation subject to the Seebeck effect and Joule heating with boundary conditions of T_H and T_C. The first term in each equation represents the Peltier effect which is reversible heat absorption or emission at the leg interfaces. The second term is the thermal resistance of the thermoelectric leg pair for the temperature gradient of the boundary surfaces. The third term is the Joule heating in the semiconductor leg pairs operating at a specific current [6]. Applying conservation of energy gives the following relationship

$$P_{TEM} = q_H - q_C = N\alpha I(T_H - T_C) - NR_e I^2 \quad (1.1.5)$$

where P_{TEM} is the electrical power extracted from the system by the thermoelectric module [4]. Although equations (1.1.3-1.1.5) describe the performance of an individual thermoelectric module subject to T_H and T_C boundary conditions, more complex formulations are required for

modeling entire thermoelectric power units utilizing multiple modules and with thermal resistance between the modules and the heat source and sink.

Several authors have developed models in the past that take into account the complexities of a thermoelectric generation system. Bethancourt, et al. performed a study and developed a model of differential equations for a thermoelectric generator in a counter flow heat exchanger configuration [7]. They looked at optimizing power output by varying many system parameters as well as some thermoelectric characteristics. Major limiting assumptions were that axial heat flow was negligible, so this was strictly a 1D analysis and heat sinks were not used, which is unlikely in real systems. Crane and Jackson performed a similar study that modeled the heat exchanger and the thermoelectric modules, solved for their performance numerically and tested them experimentally. Their application focused on waste heat recovery from an IC engine and used a cross flow heat exchanger as opposed to the counter flow exchanger that Bethancourt used. The models were then used to optimize their findings and produce a power per cost function for the system [8]. These authors as well as others have developed models for thermoelectric power generation systems but each of these models is highly specific to a system or has limited validation.

Previous work in the Rochester Institute of Technology (RIT) Sustainable Energy Lab (SEL) involved a first iteration of thermoelectric subsystem modeling. Kevin Smith, a previous graduate student, developed a model that solved a system with multiple thermoelectric modules, implemented a pricing model, and then optimized the system with relation to its cost and power output [9]. The current model predicts power recovered under some specific scenarios but has issues in other configurations. The discrepancies can be compensated for in a new model that will address specific phenomena not accounted for in the Smith model. Some of these phenomena include module voltage/current mismatch and three dimensional heat spreading.

Module mismatch reduces performance of thermoelectric modules as they are connected together electrically. The degraded performance is suspected to be a result of the difference in key module parameters between modules including internal electrical resistance and Seebeck coefficient. Modules are producing less power than predicted by simple modeling means, so improved models need to be developed and employed. Experimental data must first be gathered by comparing thermoelectric modules with varying Seebeck coefficients and internal electrical resistances to test and validate the module mismatch models. Each module has these specific parameters that affect their voltage-current relationship thus affecting their maximum power produced. If the values of the maximum power points of the modules' power curves were simply added together, it could be significantly more than a maximum power point of the power curve that represents the modules linked electrically. A model for modules linked electrically in series as well as electrically in parallel is needed to accurately predict power recovered by modules with unequal material parameters.

Three dimensional heat spreading presents itself as an issue when modules are placed into position on the heat exchanger. If the contact surface areas of the heat exchanger face and the module face differ, spreading resistance is expected to be present. The additional resistance makes it difficult to accurately predict power output using an energy balance system model. The resistance due to spreading needs to be modeled and incorporated into the overall system model. Better understanding of this effect is needed for better performance prediction and making the system model more robust.

A basic drawing of a single thermoelectric in a system with insulation surrounding it, solid surfaces holding it in place and a convective medium across these surfaces on the hot and cold side is represented in Figure 1.3. Working fluids may include water or air, both of which are currently available when testing in the Sustainable Energy Lab. The portion of the drawing labeled "surfaces" may consist of various fin systems depending on the setup. Where air flow is applied, fins are used to help with heat transfer and contact temperatures. Baffled cold plates that allow water to flow through are available for when water is used as the cold side convective medium. Figure 1.4 displays the basic thermal model of the system shown in Figure 1.3 as a schematic with heat flow and electrical power displayed.

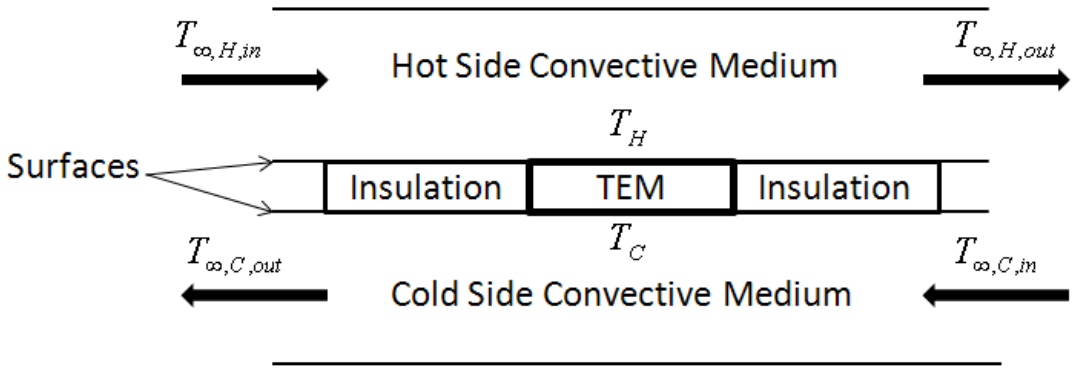


Figure 1.3: Drawing of basic thermoelectric generation system.

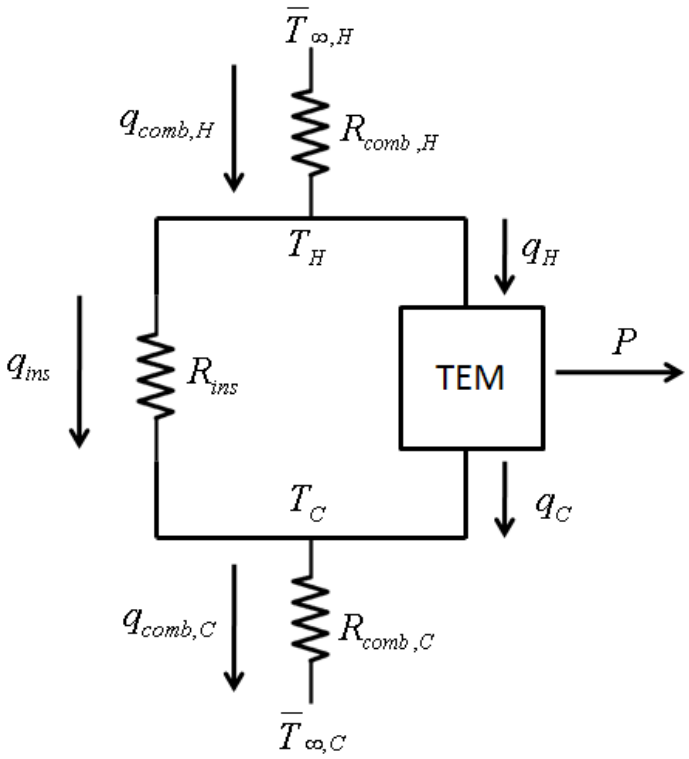


Figure 1.4: Schematic of basic thermoelectric generation system.

The flow of heat, varying temperature, and operation of the thermoelectric module can be modeled using several equations and unknowns including P, q_H, q_C, q_{ins}, $q_{comb,H}$, $q_{comb,C}$, I, T_H, T_C, $\overline{T}_{\infty,H}$, $\overline{T}_{\infty,C}$, $T_{\infty,H,out}$, and $T_{\infty,C,in}$. Some of the unknowns can be seen in Figure 1.4. This basic model developed under LaManna et al. [10] serves as the starting basis for Smith [9]. The model works under the assumption that heat flow is one dimensional and that there are no heat losses. Also, as convection occurs, the working fluid is assumed to have a linear temperature distribution from inlet to outlet [10]. P is the electrical power generated from the thermoelectric while q_H and q_C are the heat absorbed and heat rejected by the thermoelectric device, respectively. q_{ins} represents the heat that flows through the insulation where R_{ins} is the thermal resistance provided by the sheet of insulation surrounding the thermoelectric module. Insulation is used to cover the area that is not occupied by the thermoelectric module in an effort to improve the performance of the module by limiting bypass leakage. Reducing bypass leakage will maintain a higher temperature difference between the hot and cold sides. Both $q_{comb,H}$ and $q_{comb,C}$ take into account heat flowing through the convective medium, fin geometries if they are in place, and base plates for the hot side and cold side, respectively. Both R_{comb}'s take into account all of the convective and conductive thermal resistances for their respective hot or cold sides including the plates, fins, air or water convection in the system, and any insulation other than that surrounding the thermoelectric module. The temperatures T_H and T_C occur at the hot and cold surfaces of the thermoelectric, respectively. The variable, I, represents the current that develops as a result of the temperature gradient across the thermoelectric module surfaces. Each of these

variables is built into an energy balance for the proposed system. The model is further described, including its weaknesses, in later sections of this thesis. It is also developed to be more powerful and encompassing as well as have the capability to interact with TEPSS, a software tool being developed by the Rochester Institute of Technology Sustainable Energy Lab to optimize thermoelectric power systems.

The system grows much more complicated as R_{comb}'s are broken down into its actual resistances including resistance from conduction through the metal plates, fin structures, and resistance due to convection. Also, as more modules are added, the discretized model of the system becomes larger, with many more equations being solved simultaneously due to the high degree of coupling between equations.

Smith's model discretizes the heat exchanger and thermoelectric modules into a finite number of zones, performing an energy balance on each zone individually. Smith's model accurately predicted the output of power for very specific systems with low numbers of modules when compared to experimental data. However, discrepancies existed when more modules were introduced to the system, more zones were introduced to the system, and heat exchanger geometries were varied. Adding more modules and zones is believed to cause discrepancies because of module mismatch and heat spreading in three dimensions as well as issues associated with the repeatability of the experimental setup. This heat spreading also affected the model predictions from the various fin configurations [9]. These discrepancies need to be addressed to allow the model to more accurately predict a system's performance under a wide range of configurations. The higher accuracy will allow the system model to be implemented into the TEPSS simulation tool being developed at RIT to determine the feasibility of thermoelectric power systems for current and future thermoelectric materials.

1.2 Motivation

Thermoelectric devices are making gains in their efficiencies and power outputs for their most effective operating conditions. As these advances are made, applications for these devices need to be explored. Their implementation into industry could lead to a plethora of power recovered from waste heat in many different applications. Power utilities, turbine and compressor exhausts, manufacturing plants, automobile exhausts, and incinerator plants are prime examples of applications that could benefit from a thermoelectric system. Many of these systems, either directly or indirectly, use finite resources as their source of fuel. In the United States of America, about 50 quadrillion BTU's of energy is dissipated as waste heat annually [11]. Recovering 1% of this energy will equate to approximately sixteen, 1000 MW power plants. One GW of electricity will power approximately 780,000 homes based on average residential electricity consumption. Sixteen power plants could power approximately 12.5 million homes. Figure 1.5

shows the annual energy consumption for the U.S.A., the source of the energy, and allocation of the energy that is used from 2009.

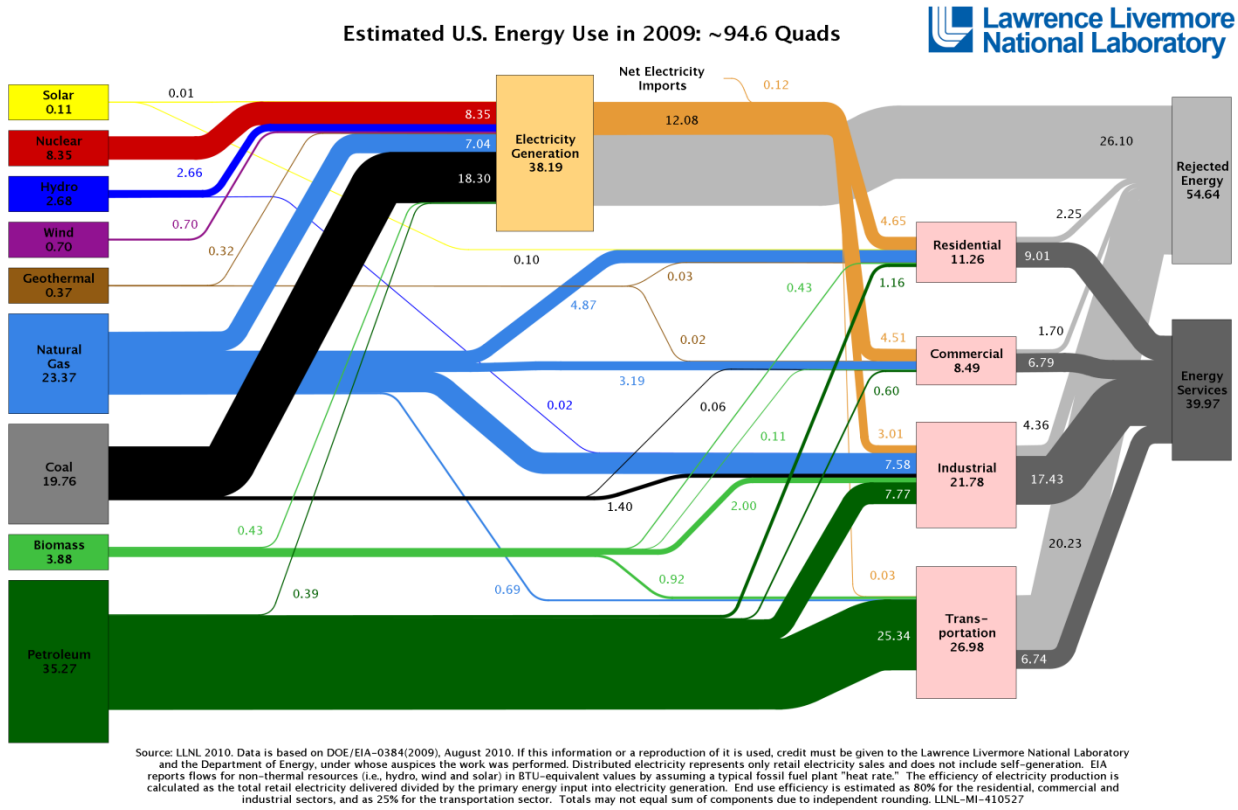


Figure 1.5: Estimated U.S. energy use in 2009 [11].

In addition to industry uses, interplanetary spacecraft propulsion as well as small, independent, or wireless systems for remote sensing applications, control equipment, and metering devices can and currently utilize thermoelectric technology [12]. An abundance of applications could surface with the advances made in technology at the module level.

For industry purposes, heat exchanger subsystem level needs to support the module level advancements to ensure proper design for the application of the thermoelectric modules. A modeling tool will allow for better development of thermoelectric subsystems for industry use and provide justification for implementation into the thermodynamic system.

A tool that is being developed by RIT SEL, Thermoelectric Power System Simulator (TEPSS), consists of a modular platform that simulates performance of a thermoelectric energy system. Performance can be optimized by varying thermoelectric parameters as well as system design parameters. Each system component has a module that interacts with the other modules and each module has specific physical and economic characteristics based on input values. Modular components have a set of input, output, and design variables. Figure 1.6 represents the optimization shell with each modular component of a simple Brayton cycle with a thermoelectric

recuperator [13]. For each system component a detailed engineering and economic model must be developed with customizable design variables.

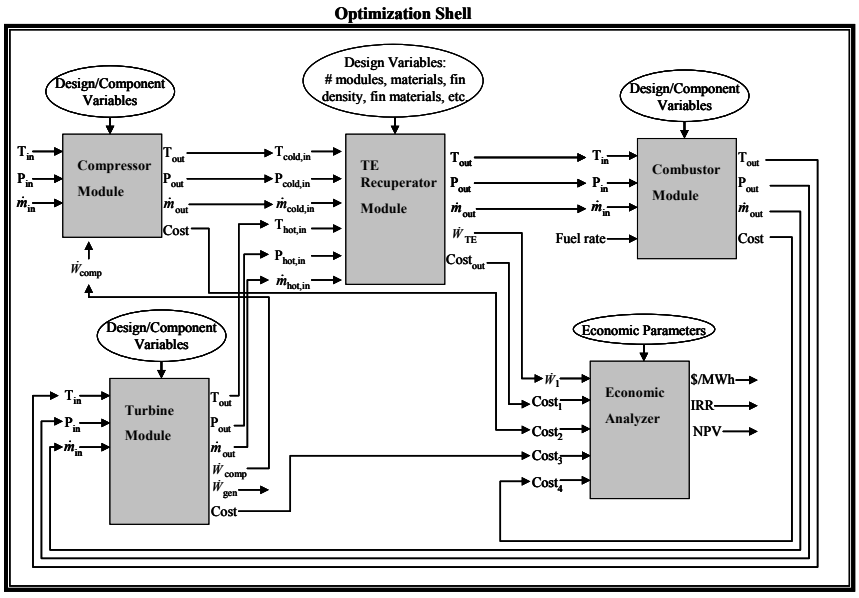


Figure 1.6: Sample TEPSS module integration for system modeling and optimization [13].

Figure 1.7 represents an enlarged image of the thermoelectric power unit component model shown in Figure 1.6 that is the focus of this thesis. It is composed of the model for a thermoelectric system taking into account the heat sinks geometry and materials, number and type of thermoelectric generators, operating conditions, and costing functions [13]. The advanced model is programmed into a simulation subsystem and serves as the key component in the TEPSS platform. This portion of the simulation environment predicts the power recovered, cost, and additional secondary output. The modeled subsystem determines the effectiveness of implementing a TE recuperator into an actual thermodynamic system.

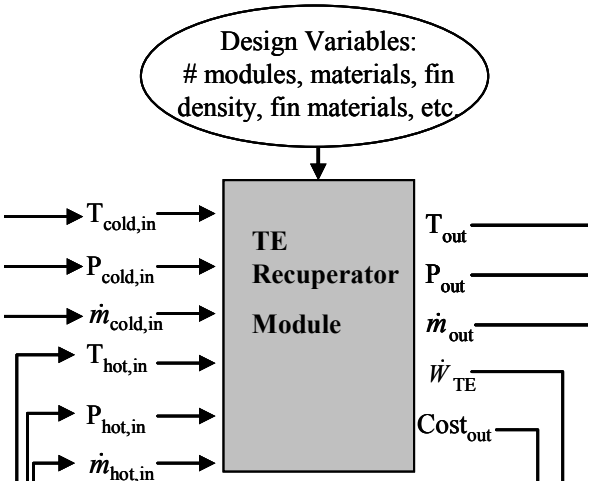


Figure 1.7: Thermoelectric Recuperator Module from TEPSS Program [13].

TEPSS has the capability to simulate a thermodynamic system with and without the proposed thermoelectric subsystem(s). This allows the user to perform a direct comparative analysis as well as obtain quantitative results. The results obtained span every point of the entire system and provide detail to the exact operation of the system. TEPSS also allows for optimization of design variables when proposing a thermoelectric heat exchanger as well as other system components. This versatility provides insight into the overall performance of a system.

1.3 Objectives

The overall goal of this thesis is to improve, validate, and implement a thermoelectric subsystem model in Thermoelectric Power System Simulator. This requires several tasks to be set and objectives to be met. These include quantifying module mismatch and experimentally validating the models, calculating the thermal resistance that results from heat spreading in three dimensions through an analytical model and experimentally verifying the analytical model, and developing a robust model to be programmed into a component for use in TEPSS and substantiate the model experimentally.

Visible in past experimental data, thermoelectric modules linked electrically in series or parallel have not generated the expected electric power output. The potential reason for this discrepancy in predicted and actual power is mismatch of thermoelectric device parameters [9]. Similar mismatch losses occur in photovoltaic cells and the operation and theory of photovoltaic cells is comparative to thermoelectric modules. To test the mismatch theory, an equation that predicts the power output of a series of thermoelectric generators will be developed as well as an equation for parallel thermoelectric generators. The model will be developed to quantify module mismatch by comparing the ratio of peak power of coupled thermoelectric modules to the sum of the peak powers of individual modules for modules in series and parallel configurations given the ratios of mismatch in module Seebeck parameters and internal electrical resistances. Series and parallel mismatch power ratio models will be experimentally validated. Forty modules will be tested individually to characterize their constant parameters including Seebeck coefficient and electrical resistance. The characterized modules with known properties will then be used together to demonstrate the mismatch effect. The experimental data will be compared to the theoretical predictions for validation.

A model will be developed for finding an effective thermal resistance as a result of heat spreading in three dimensions due to the non-unity aspect ratios of the thermoelectric modules to their extended surfaces. This will be done by adapting or modifying an existing model found in existing mathematical literature. The calculated analytical solutions will be compared to numerical solutions and then compared to experimental results. Existing thermoelectric system models use finite element analysis to predict the heat conduction through the fins and apply it to the solution. This provides a major limiting factor on the application of the thermoelectric system

model for the fin arrangement. Using only a 1D conduction assumption for analysis of fin resistance does not provide accurate results. Therefore, a 3D heat spreading conduction needed to be added to the model. The finite element analysis is representative of only predefined fin geometries and does not allow for optimization of a general fin structure. This removes a key ability of TEPSS which is to adapt selected parameters to the system conditions and provide a best result. Thermal spreading of heat from the thermoelectric through the fins will be evaluated using a specific model that will take input size parameters, non-dimensionalize them, and provide a three-dimensional spreading resistance, in lieu of an finite element 3D spreading resistance analysis, to be added to the one-dimensional resistance already accounted for. Using concepts and mathematical procedures developed by Ellison [14], thermal spreading resistances can be determined without the need for a finite element analysis (FEA). The analytical model will be compared to numerical results provided by use of the ANSYS® software. This will allow for direct comparison of variation in the shape, size, and number of the current fins. After the model is functioning and producing results similar to select FEA setups, experimental verification will be done. The experimental approach will require several samples of aluminum blocks with known properties to be tested in the power unit to provide verification of the predictions made by the model. The aluminum blocks vary in orientation, size, and number to test the limits of the heat spreading model.

Adapting an existing thermoelectric system model and further developing the model into a single component in MATLAB® to be used in the open source TEPSS environment is a priority of this research. The newly developed thermoelectric model will be expanded to incorporate several different methods for inputting module performance parameters. The methods to be included are using measured performance data on existing modules to provide key parameters, extracting module performance characteristics from thermoelectric cooler (TEC) and thermoelectric generator (TEG) manufacturer datasheets, finding the key parameters by modeling module performance based on geometric and material properties of proposed thermoelectric devices. The lattermost method will be based on work by Sandoz-Rosado [15]. Several different fin type configurations for selection by the TEPSS user are to be modeled including rectangular straight fins, offset strip fins, and aligned and staggered pin fins. Hot side and cold side fins will have the option to be set to different types of finned geometries with adjustable design variables to customize the heat exchanger. Using the newer more robust thermoelectric heat exchanger model, the MATLAB® code can be adopted by the TEPSS project to analyze the feasibility in specific applications of thermoelectric systems and optimize selected design variables to determine optimal performance related to cost. The overall goal is to develop a model that more accurately predicts power output of a thermoelectric subsystem operating under the above mentioned parameters and then implement the model into a module to be used as part of the architecture of a full scale feasibility study. The subsystem module to be used in the full scale study is a smaller part of a large network of computer code developed for the New York State Energy Research and Development Authority. The component model will be developed allowing for the optimization of the overall system by varying module level design parameters. The model

needs to be experimentally validated under a wide range of scenarios accounted for in the model. Major parameters examined include flow rates and operating temperatures. The experimental validation will be used to evaluate the improved model. If discrepancies still exist, potential issues will be identified and improvements will be recommended for future work. If the experimental results and improved model predictions closely correspond, then the model will need no further improvements.

2 Literature Review

2.1 Module Level

As described in Chapter 1, a thermoelectric module has the ability to generate power and provide a source of heating and cooling. The driving principle for thermoelectric power generation comes from the Seebeck effect. It is often assumed that the Seebeck coefficient is constant for each leg and can be found using a couple methods of averaging. In reality, the temperature varies over the length of the leg between the hot side and cold side. This means that the Seebeck coefficient is a function of temperature. A mean value can be found for the leg by averaging both the hot and cold side Seebeck coefficients. The average Seebeck coefficient for a leg can also be found using the integral average as seen in equation (2.1.1) [2].

$$\alpha = \frac{1}{T_H - T_C} \int_{T_C}^{T_H} \alpha dT \tag{2.1.1}$$

By averaging the Seebeck coefficient for each leg type in the thermoelectric module, the Thomson effect is accounted for and built into the Seebeck coefficient. Most authors assume the Thomson effect to be negligible when in reality; it is accounted for in the averaging of the variation in the Seebeck effect. The Thomson coefficient is denoted by τ and is seen in equation (2.1.2).

$$\tau = T \frac{d\alpha}{dT} \tag{2.1.2}$$

This can be built into the power equation by multiplying the Thomson effect times the current and the temperature difference between the hot and cold side of the legs. It is only needed to be accounted for when there is a hot side Seebeck coefficient and a cold side Seebeck coefficient. Overall, it is safe to assume that it is negligible as it is accounted for when the Seebeck is found as an average of the leg.

To develop the thermoelectric effect, a positive carrier or p-type semiconductor and a negative carrier or n-type semiconductor must be joined in electrical series. Figure 2.1 displays a thermocouple with a load resistance of R_o across it.

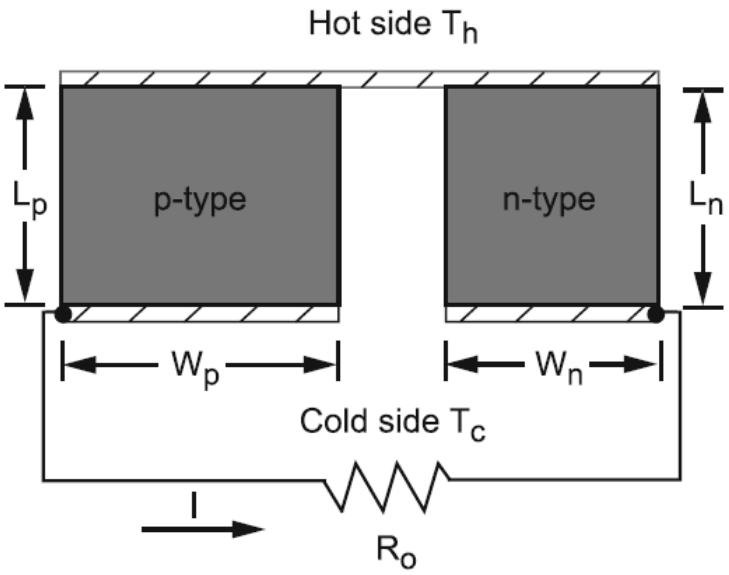


Figure 2.1: Thermocouple with a p-type and n-type junction [16].

To solve for the Seebeck effect, thermal conductivity, and electrical resistance, the heat diffusion equation must be looked at and represented in equation (2.1.3).

$$\frac{d^2T}{dx^2} + \frac{I^2\rho}{kA^2} = 0 \qquad (2.1.3)$$

This expression works under the assumption of steady state, one dimensional, constant conductivity, and uniform heat generation from Joule heating. The second term represents the Joule heating where I is the current, ρ is the electrical resistivity, k is the thermal conductivity, and A is the cross sectional area of the thermoelectric pellet. Joule heating increases with greater electrical current or resistivity and decreased thermal conductivity or cross sectional area. The boundary conditions for differential equation are $T=T_C$ at $x=0$ and $T=T_H$ at $x=L$. The "*Cold side*" is defined as $x=0$ while the "*Hot side*" is defined as $x=L$. The solution to this equation is expanded to p-type and n-type pellets with different properties in section 5.2.2 where a more detailed explanation is shown.

Providing values that describe the properties of the thermoelectric module can be applied to the figure of merit as described by equation (1.1.2). Figure of merit, represented by ZT, can be used to determine the efficiency of a thermocouple. This is actually a maximum efficiency when the two surfaces are held constant at T_H and T_C. In real systems, T_H and T_C could not be held constant for any current which makes this just a theoretical limit. The maximum efficiency is described by equations (2.1.4) and (2.1.5) [2].

$$\eta = \frac{T_H - T_C}{T_H} \times \frac{M-1}{M+T_C/T_H} \qquad (2.1.4)$$

$$M = \sqrt{1 + \frac{1}{2} Z(T_H - T_C)} \quad (2.1.5)$$

The first portion of equation (2.1.4) is the Carnot Efficiency where the second half contains the average *ZT* within the module and accounts for irreversibilities. T_H and T_C are the absolute temperatures of the hot and cold sides of the semiconductor legs, respectively. As mentioned previously in Chapter 1, there is a great focus by researchers to increase the figure of merit to approximately 3 to drive modules to an efficiency that could benefit a wider range of applications [3]. Researches also focus on developing models of heat exchangers to better suit the needs of the modules and prepare for the day that the efficiency is to a level worth investing in across the wider range of applications.

2.2 Heat Exchanger Level

There are many mathematical models and approaches for simulating a heat exchanger that utilizes thermoelectric modules for power generation. Each of these models applies general heat transfer techniques in cohesion with thermoelectric module equations for a system level analysis. Authors of these models often build on each other but determine their own course of development sometimes including or leaving out specific aspects of the science due to assumptions. Several models are reviewed with a description of their features and approach.

One of the earliest model was developed by Bohn model in 1981 which is an extension of the effectiveness-number of transfer units (ε-NTU) method for heat exchanger analysis [17]. The model implements thermoelectric generator equations into the common ε-NTU equations by setting up a dimensionless ratio of actual power generated to the maximum possible power generation. The ε-NTU method linearizes the function of temperature difference to make the system solvable while equations describe how well the heat exchanger area is being utilized. The heat exchanger has TEG's located in the wall separating hot and cold fluid streams. Fluid properties, flow rates, flow configuration, and thermoelectric generator properties are provided to the model. The thermal resistances from the fluid to the thermoelectric junctions must be provided which means separate calculations from the model are needed. These resistance calculations should include fin efficiencies, thermal conductivity of heat-exchanger plate, and thermal contact resistances. An example case is provided for a parallel flow configuration with promising results, however, there is no experimental testing to validate the model. Also, no costing function is proposed for this system.

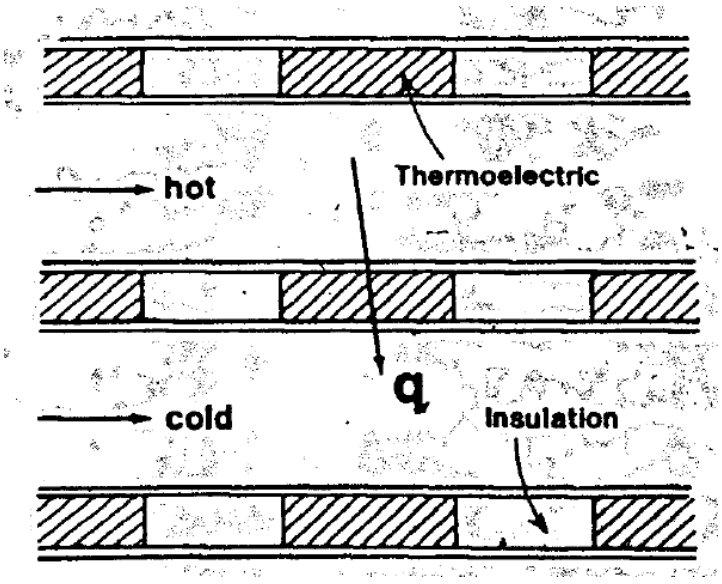


Figure 2.2: Detail of heat exchanger and thermoelectric generator combination [17].

The Esarte, Min, and Rowe model also uses ε-NTU method to simplify calculations but is not entirely an NTU model [18]. Esarte, et al. set up an energy balance with a hot and cold side heat exchanger with a thermoelectric module in between. A thermal equivalent circuit is developed for the overall heat transfer coefficient taking into account fluid boundary layers, heat exchanger bases, paste layers, and TEM thermal resistances. A parallel flow heat exchanger is modeled with a log-mean temperature difference provided for the temperature at the module junctions. The log-mean temperature is eliminated using the NTU method and getting it in terms of inlet temperatures, mass flow, overall heat transfer coefficient, and thermoelectric properties. Limited experimental results are provided and no costing function is provided.

Bethancourt, Echigo, and Yoshida [7] used numerical computations to model a thermoelectric generator in a counter flow heat exchanger configuration because it had not been done before. They assume one-dimensional analysis (axial conduction in channels and partition walls is negligible), extended surfaces were not used, thermophysical properties are held constant, and mass flow rates in both the hot and cold channels are set equal for simplification of the problem. The numerical approach is used to solve the problem because of the non-linear system of equations. The model was used to determine that maximum power and system efficiencies with respect to specific parameters through optimization. No costing function is used and experimental testing is not performed.

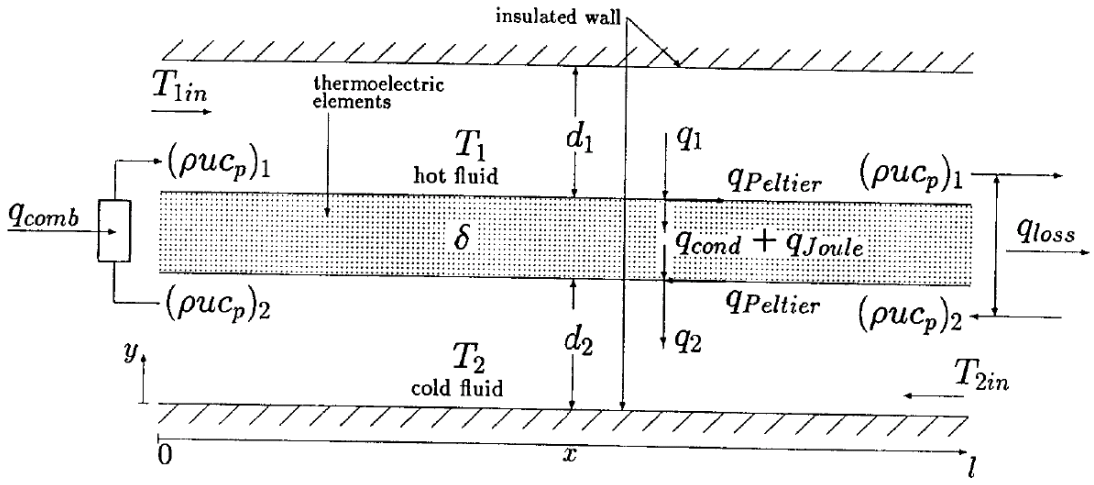


Figure 2.3: Physical model under consideration [7].

Hendricks and Lustbader developed a thermoelectric/heat exchanger system analysis tool for National Renewable Energy Laboratory [19]. The model was used to investigate the behavior of important TE and thermal system design parameters. Their model focuses on how heat exchanger performance, parasitic heat loss effects, and thermal interface resistance effects are directly and simultaneously coupled with TE device performance for analysis of vehicle and industrial applications. The model predicts potential TE system power output using a set of twelve equations that are a function of the hot and cold side module junction temperatures with known inlet temperatures. For design optimization, hot and cold side heat exchanger effects, parasitic heat losses, electric contact resistances, and thermal interface resistances are accounted for. Optimization is a strong focal point of this model but no experimental testing is performed to verify the analysis.

A thermoelectric generation system for an automotive exhaust and engine coolant is developed by Karri from Clarkson University [20]. The system model utilizes known values including gas inlet temperature and flow rate, coolant inlet temperature and flow rate, and external load resistance. The TEG system finds the exhaust gas and coolant outlet temperatures, thermoelectric module junction temperatures, the power generated, and the pressure drop across the exhaust and coolant heat exchangers. Assumptions of Karri's model include perfectly insulated regions between the TEMs as well as the exteriors and sides of the heat exchangers. It also assumes negligible thermal resistance to heat flowing in the normal plane of the thermoelectric modules resulting in a 1-D heat transfer assumption. The TEG system is divided into four sections with the same temperature profile. All of the system equations are combined into four equations with four unknown temperatures and this setup is solved through an iterative process using the Newton-Raphson method. Pressure drops and heat transfer coefficients are calculated for the heat exchangers to support the analysis. Experimental validation showed reasonable predictions made by the model which resulted in the recommendation that the heat lost to the ambient and

the non-uniform distribution of flow rate be fixed. There was a sensitivity analysis performed to target key parameters and there was no focus on economic functions.

Additional work was performed by the researchers at Clarkson University including several case studies. The case studies utilized the Karri model as well as the NREL Advanced Vehicle Simulator, or ADVISOR. Karri, Thacher, and Helenbrook [21] used ADVISOR to simulate the whole system where the thermoelectric generator system was added to the code. This piece of work examines two cases: a Sports Utility Vehicle (SUV) and a stationary compressive natural gas fueled generator. An energy balance was performed with the thermoelectric governing equations and the heat exchanger equations which came from another piece of work. This system solved six system equations simultaneously using a multi-dimensional Newton-Raphson method to find the four unknown temperatures in each of the heat exchanger sections. There were sixteen proposed sections for the heat exchanger to allow greater power generation. It was concluded an increased efficiency was needed in order to make the technology viable. Other case studies were performed through Clarkson personnel [22] with similar results. Once again, ADVISOR was used with the Karri model. The end goal was to convert thermal energy from engine exhaust into electrical energy and publish the results. The library from ADVISOR included an SUV, a stationary electrical generator powered by natural gas, and a hybrid engine-powered transit bus which were all used in the case study. The results of each of the studies were that the efficiency must be higher, the overall heat transfer coefficient for the coolant and exhaust heat exchangers should be high, and the loss ratio must be low meaning that coolant pumping power needs to be at a minimum and the exhaust heat exchanger needs to be reduced to avoid parasitic losses.

Suzuki and Tanaka provide energy balances for varying systems of large-scale flat panels that exhibit thermoelectric generation capabilities [23]. Each of these panels is exposed to a hot and cold thermal fluid. A non-dimensional analytical approach is taken when solving for the temperature profiles and ultimately the output power. The non-dimensional functions allow for focus on system design. A matrix is set up to represent the temperatures between panels and the thermophysical properties of the thermoelectric materials and the working fluids. The matrix solution method can represent various types of systems as defined by the boundary conditions. Possible systems include parallel flow, counter flow, isothermal, and a mixed flow arrangement for multi-panel systems. In addition, there are fluid flow path shapes including meandering type, helical type, and branched type. These allow for many combinations as the number of panels is increased. As the energy balance is performed, calculated fluid temperatures are used to find the thermoelectric junction temperatures, which are the temperatures of the hot and cold side of the thermoelectric module. The junction temperatures are then used to find power generated. This solution method and model is expanded to cylindrical multi-tubes which is very similar but relies on a radial coordinate system [24, 25].The overall solution method and theory of this body of work provides excellent methods for solution. However, this is based in theory and lacks a practical system setup as it focuses on unrealistic thermoelectric panels instead of a heat exchanger and TEG setup. Application of this solution technique to an experimental

configuration would provide greater insight to the capabilities of this energy balance model and solution method.

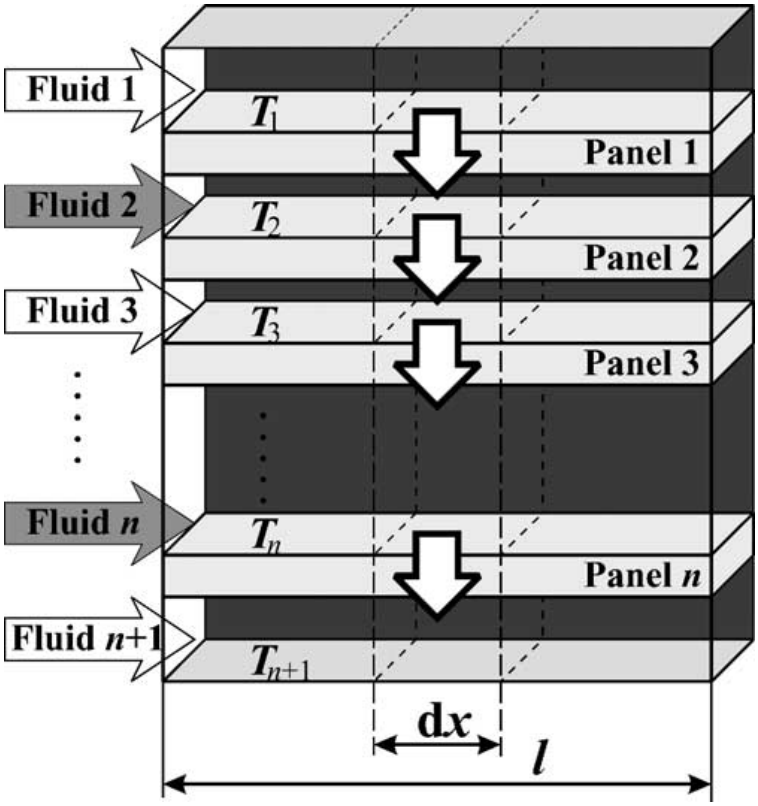


Figure 2.4: Thermoelectric panels and heat transfer fluids illustration for generic system configuration [23].

Crane and Jackson investigate thermoelectric waste heat recovery with regards to cross flow heat exchangers [8, 26]. A focus is placed on optimization with regard to overall system performance. Power losses from an air fan which provides cooling and from a fluid pump which provides heated liquid are taken into account. A cross flow heat exchanger model is validated against measured performance of advanced cross flow heat exchangers without thermoelectrics. The focus of the optimization is to simultaneously optimize the heat exchanger geometry and the TE geometry while including penalties for the hot side and cold side pressure drops. A typical thermoelectric model is used that accounts for conduction, Seebeck effect, Joule heating, and contact resistance. Thomson effect is neglected. Also, it is assumed that the mid-plane boundary is adiabatic. The numerical model was implemented in MATLAB® with the primary discretization along the axial direction of the hot fluid flow. For the cool air side cross flow, a log mean temperature difference is used at each axial location. An iterative Newton-Raphson method was used which utilized an analytically calculated Jacobian matrix. The numerical simulations were compared to experimental data with good agreement between them. The optimization study was conducted with the validated Crane and Jackson model to show the potential use of cross flow heat exchangers for an internal combustion engine with a thermoelectric based waste heat

recovery. The results showed that a net power output of 1 kW could be achieved but the power per cost ratio could be as high as 1.1 kW per $10,000.

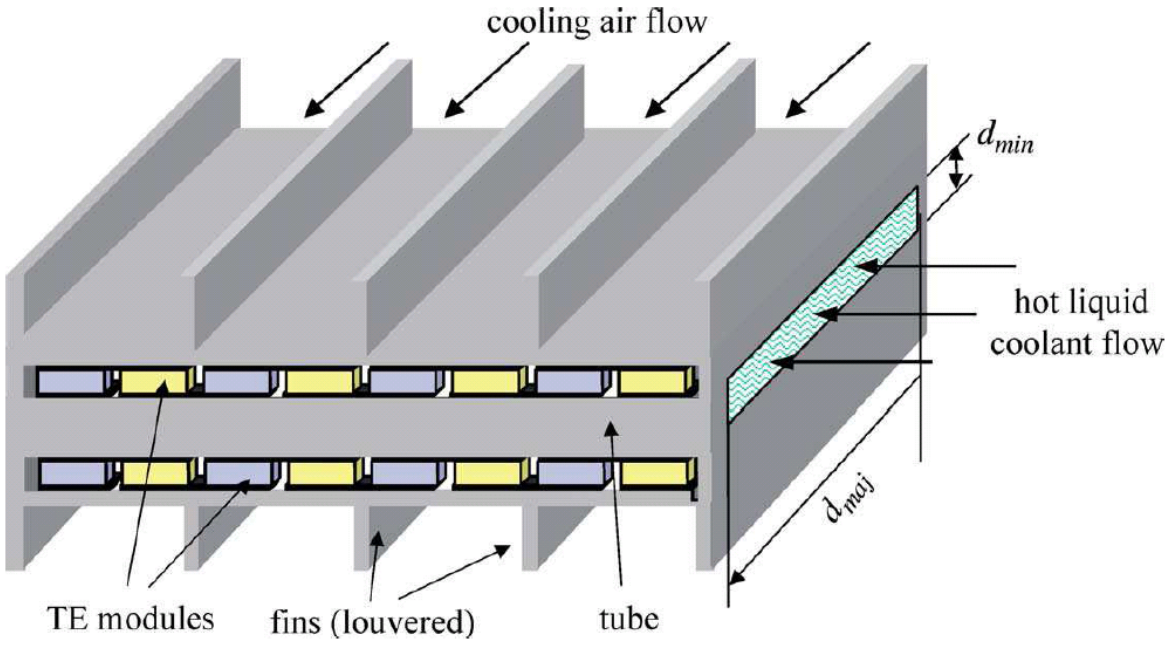


Figure 2.5: Cross flow heat exchanger for thermoelectric waste heat recovery [8].

Yu and Zhao developed a numerical model for prediction of performance of a thermoelectric generator with a parallel-plate heat exchanger [27]. It is assumed that the flat thermoelectric modules are held tightly between hot and cold fluids which have multiple thermocouples with a single layer of P-type and N-type semiconductors. The thermocouples along the fluid path are considered to be connected electrically in series. The authors also assume that axial heat conduction in the thermocouples is negligible; the thermal resistance through the heat exchanger plates, ceramic plates, and metallic strips of the thermoelectric modules are lumped together into an equivalent resistance which neglects thermal contact resistance; heat losses between the modules and plates are ignored; and the space between thermocouples is ignored. A typical energy balance is used to set up the model with the differential equations discretized along the axial direction of the hot fluid. The solution to the numerical model is provided using an iterative method. Simulations were performed to study the effects of the various parameters. An experimental study based on the model from Yu and Zhao [27] was performed by researchers Niu, Yu and Wang [28]. A comparison of the experimental results with the numerical model is presented in this work. A two fluid, multi-plate, multi-pass, counter/parallel flow heat exchanger with thermoelectric generators was created for the experimental phase. The data obtained through experimentation shows that the numerical model over predicts performances of the TEG's over the entire range of data. At lower temperatures, the model displays better agreement with experimental results, but as hot fluid inlet temperatures are increased, the prediction

diverges from the measured values. The discrepancy is credited to the lack of accounting for heat losses and the fact that the thermoelectric properties are treated as constants.

Overall, each of these authors has significant attributes in their work which serve as a sound foundation for additional research. Most of the work described above performs energy balances and utilizes the basic governing equations for thermoelectric generation. The variations between each of the models are the solution technique and the limiting assumptions that are made. Solution techniques vary from ε-NTU method for heat exchanger analysis to other analytical approaches or numerical solution techniques. Each has its strengths and weaknesses. More importantly, the limiting assumptions that are made have drastic effects on the models. Neglecting losses to the environment and contact resistances could likely distort solutions to be more favorable than in reality. Unrealistic adiabatic regions will definitely contribute error to model predictions. Sometimes thermoelectric modules were considered to be far more capable of generation than the modules that exist in reality. This type of analysis provides a good source of information for parametric studies but doesn't provide useful, realistic data. Other assumptions that are made that can provide favorable results that might not be based in reality include constant properties throughout heat exchanger systems, heat transfer coefficient value or model, and one dimensional analysis of thermoelectric generators. Some authors assume peak power loading or peak efficiency, neither of which may not be true under the modeled conditions. From an examination of the literature it is evident that there is considerable room for improvement in modeling of heat exchangers that utilize thermoelectric generators for heat recovery.

2.3 Heat Spreading

Several methods for quantification of heat spreading are considered. These include analytical, numerical, and experimental solution methods. The use of experimental solutions to determine a value for heat spreading phenomena is not ideal for a TEPSS component model because TEPSS and the thermoelectric power unit model share the goal of being applicable to many system configurations. A numerical solution is currently used in the research performed under [9]; this process requires too much computational power to run quickly and effectively and would not be supported by the optimization routine that is integral to the TEPSS package. Therefore, numerical approaches are unlikely to provide a practical means of estimating the rate of heat spreading. Analytical techniques have been developed in the heat sink literature which can be adopted and used to estimate heat spreading. Analytics are the ideal solution method because of their ability to be used in numerous applications and situations.

Three dimensional models can be applied to estimate heat spreading. Green's function is used by Ellison [14] for conductive heat transfer from a planar heat source to a surface of larger cross sectional area. When this heat source is not of the same aspect ratio as the mated surface, heat spreading will occur in three dimensions.

The model developed by Ellison analytically determines an exact effective thermal resistance [14]. It also has the ability to separate this effective resistance into components that represent a one-dimensional resistance and a spreading resistance. The steady state heat equation is applied in three dimensions with boundary conditions that define most of the geometry as adiabatic. A convective boundary condition is applied to the side opposite of that with the source of heat energy. The heat flux at the source is assumed to be uniform and the surface on which it exists is taken to be adiabatic. Also, the heat transfer coefficient is presumed to be constant across the plane opposite the source plane. Both the thermal conductivity and the heat transfer coefficient are to be independent of temperature. The analysis is for flat plates and several resistance plots are developed for various dimensionless inputs derived from the dimensioned geometrical parameters. This method can be extended to thermoelectric modules placed on finned heat sinks. An effective convective coefficient based on an effective area is needed which essentially turns the finned heat exchanger into a flat plate. Extensive solutions are not explored or provided by Ellison [14] for finned heat exchangers, but the model provides opportunity for their development.

The Ellison model provided methods for several authors to expand upon [29]. Like Ellison [14], Rhee and Bhatt [29] provide a three dimensional solution to the heat conduction equation using Green's functions. The model presented is used for similar geometries with a small square heat source on an adiabatic surface with convection occurring on the opposite face of the geometry. Figure 2.6 is a visual representation of the boundary conditions that are common to both Ellison [14] and Rhee and Bhatt [29]. The solution is provided using non-dimensional parameters similar to the research provided by Ellison [14]. The primary difference between methods is the transient functionality of the Rhee and Bhatt model. The steady state aspect of their model is compared to Ellison's and found to closely for all cases examined. The transient aspect of the model is benchmarked by applying it to a lumped sum for which a solution is also determined using a capacitance model. The two solutions are compared and excellent agreement is found between solutions.

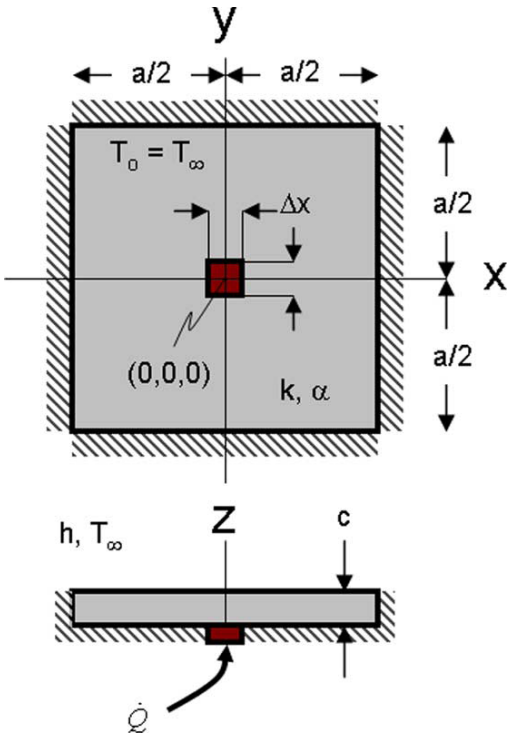


Figure 2.6: Geometry and boundary conditions modeling conduction spreading resistance [29].

The use of transient models may be of importance to some heat exchanger designers. However, the focus of this thesis and the TEPSS program is to provide solutions to steady state systems. The Ellison model will suffice for analytically determining the thermal resistance of heat spreading in a thermoelectric power unit.

2.4 Module Mismatch

The issue of module mismatch is relatively unaddressed by the thermoelectric community. Very little literature exists with regard to developing formulae to predict the mismatch effect. This portion of research has significant opportunities in developing a useful tool for thermoelectric generation system designers.

One group of scientists has taken an interest in the module mismatch challenge and brought forth methods for minimizing the effect. Researchers [30, 31, 32] have looked into mismatch and developed a Maximum Power Point Tracking (MPPT) system. The MPPT power conditioner consists of a Buck-Boost converter, internal power supply, and microcontroller. This system allows losses to be reduced but is particularly useful for the system as it experiences transients. It is less useful when the thermoelectric generators are in steady state but still has the ability to cause the TEGs to operate at their peak power. This team [30, 31, 32] offers no analytical analysis for finding the source of the mismatch or a predictive method for compensating for

them. Their device does provide a great tool for experimental methods and potential field implementation of a generation system.

Photovoltaic (PV) arrays operate in a similar fashion as thermoelectric generators. PV arrays are disputably subject to mismatch losses and should be considered in the development of the thermoelectric module mismatch predictive equations. According to Chouder and Silvestre [33], mismatch has been identified as one of the most important causes of power losses in photovoltaic applications. The work of Chouder and Silvestre [33] is to provide experimental and modeling results on mismatch effects in PV modules with a focus on the generation losses. Power losses around 10% were observed in their study associated with mismatch between PV modules forming the PV array.

A similar study to Chouder and Silvestre [33] was performed by Picault et al. [34]. Their belief is that module mismatch is caused by two main reasons: dispersion of electrical properties and non-uniformity PV cell illumination. They developed several connection schemes for each of the PV modules and analyzed the mismatch losses for each of the connection schemes. Their simulation results compared to experimental results yielded several discoveries. They found that a considerable effect in mismatch losses is caused by partially shaded PV modules as expected. However, un-shaded modules provided only 1-2% discrepancies between the experiment and simulation with regards to mismatch losses.

This research provides contradictory evaluations of the effects of module mismatch in PV arrays. Because of the similarities between photovoltaics and thermoelectrics, parallels can be drawn with regards to performance. Further exploration is needed to determine how significant the module mismatch effect is for thermoelectric modules. A method for quantifying the mismatch is needed as no known model exists in the research community as of yet.

3 Module Mismatch

Understanding discrepancies in predicted power to actual power recovered when linking thermoelectric modules electrically in series or parallel is of importance. System tests have shown a reduction in power generation potential when parameters of electrically linked modules differ. This discrepancy is known as module mismatch. Modules produce less power than predicted by simple modeling means so improved models are developed. The over-prediction will be compensated for in an addition to the steady state 1-D model that is currently being used. Experimental data is gathered by comparing thermoelectric modules with varying Seebeck coefficients and internal electrical resistances to confirm the module mismatch models. Each module has these specific parameters that affect their voltage-current relationship, thus affecting their maximum power produced. As can be seen in Figure 3.1, the voltage-current curves are different for the two modules as a result of their parameters. When electrically linking these modules, the voltage-current curve changes to represent the new properties of the module. Figure 3.2 displays the two individual modules' power curves. If the values of the maximum points of these curves were simply added together, it would be significantly more than the maximum point of the power curve that represents the modules connected electrically in series.

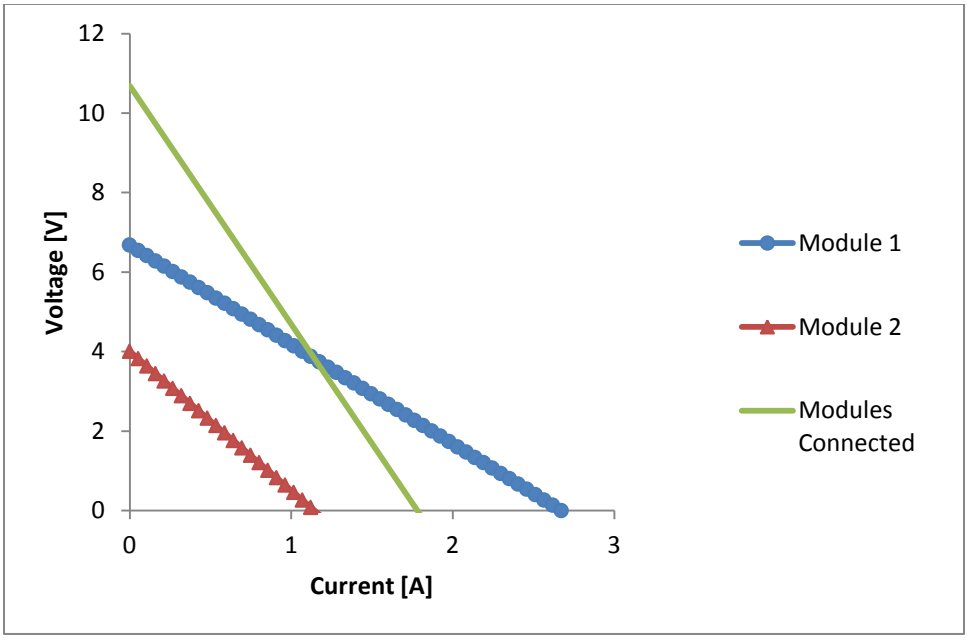


Figure 3.1: Example of a voltage-current plot for two modules.

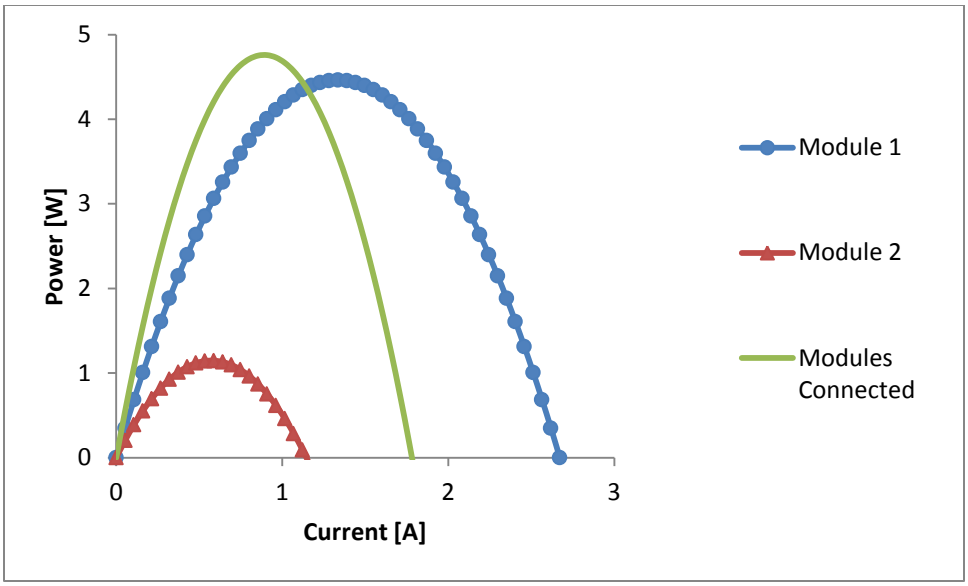


Figure 3.2: Example of a power plot for two modules.

The power, P, of a module as a function of current is given by equation (1.1.5) [4].As a reminder, $N\alpha$ represents the module level Seebeck coefficient, NR_e is the electrical resistance of the module, T_c is the hot side of the module temperature, T_c is the cool side of the module temperature, and I represents the current

$$P_{max} = \frac{(N\alpha)^2(T_h - T_c)^2}{4(NR_e)} \tag{3.0.1}$$

Equation (3.0.1) is determined by taking the derivative of (1.1.5) with respect to current and setting it equal to zero yielding the current at the maximum power point as seen in equation (3.0.2).

$$I_{max} = \frac{N\alpha(T_h - T_c)}{2(NR_e)} \tag{3.0.2}$$

Taking Ohm's Law and applying it to thermoelectric modules internal resistance produces

$$I = \frac{V}{NR_e} \tag{3.0.3}$$

Substituting equation (3.0.3) into (1.1.5) yields power as a function of voltage (3.0.4).

$$P = N\alpha(T_h - T_c)\frac{V}{NR_e} - \frac{V^2}{NR_e} \tag{3.0.4}$$

Taking the derivative of power with respect to voltage in equation (3.0.4) and setting it equal to zero gives the voltage at the maximum power point.

$$V_{max} = \frac{N\alpha(T_h - T_c)}{2} \tag{3.0.5}$$

Multiplying (3.0.2) and (3.0.5) gives (3.0.1) which serves as the relationship of the maximum power for one thermoelectric module.

Adding two modules, with different Seebeck coefficients and electrical resistances, maximum power relationships together provides

$$P_{TE1} + P_{TE2} = \frac{(N\alpha_1)^2(T_h - T_c)^2}{4(NR_{e1})} + \frac{(N\alpha_2)^2(T_h - T_c)^2}{4(NR_{e2})} \tag{3.0.6}$$

Finding a common denominator and adding the new numerators together yields

$$P_{TE1} + P_{TE2} = \frac{(NR_{e2})(N\alpha_1)^2(\Delta T)^2 + (NR_{e1})(N\alpha_2)^2(\Delta T)^2}{4(NR_{e1})(NR_{e2})} \tag{3.0.7}$$

Implementing a ratio for each parameter observed allows for simplication of the overall equation.

$$C_1 = \frac{N\alpha_2}{N\alpha_1} \tag{3.0.8}$$

$$C_2 = \frac{NR_{e2}}{NR_{e1}} \tag{3.0.9}$$

Equation (3.0.8) is the ratio of the Seebeck coefficients for two modules and equation (3.0.9) is the ratio for the internal electrical resistances for two modules.

The simplified equation with the ratios implemented is

$$P_{TE1} + P_{TE2} = \left(\frac{(N\alpha_1)^2(\Delta T)^2}{4(NR_{e1})}\right)\left(\frac{(C_1^2 + C_2)}{C_2}\right) \tag{3.0.10}$$

Equation (3.0.10) will later be used to set up ratios for the mismatch when modules are connected in series and parallel.

3.1 Governing Equations

When dividing equation (1.1.5) by the current, a relationship for voltage can be determined as a function of current given by equation (3.1.1).

$$V = N\alpha(T_h - T_c) - NR_e I \tag{3.1.1}$$

Examination of equation (3.1.1) shows that it is linear with respect to current. The electrical resistance is the negative slope and the y-intercept is given by the Seebeck coefficient times the temperature difference.

Rearranging equation (3.1.1) to solve for current as a function of voltage provides

$$I = \frac{N\alpha(T_h - T_c)}{NR_e} - \frac{V}{NR_e} \quad (3.1.2)$$

Equations (3.1.1) and (3.1.2) serve as the governing equations for modules connected in series and parallel.

3.2 Series

When thermoelectric modules are connected in series, they operate under the condition of equal current and increasing voltage. Establishing that the current is the same between modules and that there are two modules, governing equation (3.1.1) can be written as

$$V_1 = N\alpha_1(T_h - T_c) - NR_{e1}I_s \quad (3.2.1)$$

$$V_2 = N\alpha_2(T_h - T_c) - NR_{e2}I_s \quad (3.2.2)$$

for each module. Adding (3.2.1) and (3.2.2) together provides

$$V_1 + V_2 = \Delta T(N\alpha_1 + N\alpha_2) - (NR_{e1} + NR_{e2})I_s \quad (3.2.3).$$

To simplify this equation

$$V_1 + V_2 = V_s \quad (3.2.4)$$

so

$$V_s = \Delta T(N\alpha_1 + N\alpha_2) - (NR_{e1} + NR_{e2})I_s \quad (3.2.5).$$

Rearranging (3.2.5) with current as a function of voltage for series modules gives

$$\mathrm{I_s} = \frac{\Delta T(N\alpha_1 + N\alpha_2)}{(NR_{e1} + NR_{e2})} - \frac{V_s}{(NR_{e1} + NR_{e2})} \quad (3.2.6).$$

Maximum power voltage can be derived by finding open circuit voltage of equation (3.2.5) and dividing by two. Maximum current can be determined by finding short circuit current of equation (3.2.6) and dividing by two.

$$V_{s,max} = \frac{V_{oc}}{2} = \frac{\Delta T(N\alpha_1 + N\alpha_2)}{2} \quad (3.2.7)$$

$$I_{s,max} = \frac{I_{sc}}{2} = \frac{\Delta T(N\alpha_1 + N\alpha_2)}{2(NR_{e1} + NR_{e2})} \quad (3.2.8)$$

Maximum power is equal to the maximum power current multiplied by the maximum power voltage.

$$P_{s,max} = I_{s,max} \times V_{s,max} = \frac{(\Delta T)^2(N\alpha_1 + N\alpha_2)^2}{4(NR_{e1} + NR_{e2})} \quad (3.2.9)$$

Substituting ratios for the Seebeck coefficient (3.0.8) and internal electrical resistance (3.0.9) and simplifying yields

$$P_{s,max} = \frac{(\Delta T)^2 (N\alpha_1)^2 (1+C_1)^2}{4(NR_{e1})(1+C_2)} \tag{3.2.10}$$

To determine the difference between maximum power of two modules in series compared to simply adding the same two modules maximum powers together, another ratio is examined. Dividing equation (3.2.10) by equation (3.0.10) creates a ratio indicating the level of mismatch.

$$\frac{P_{s,max}}{P_{TE1}+P_{TE2}} = \frac{\frac{(\Delta T)^2 (N\alpha_1)^2 (1+C_1)^2}{4(NR_{e1})(1+C_2)}}{\left(\frac{(N\alpha_1)^2 (\Delta T)^2}{4(NR_{e1})}\right)\left(\frac{(C_1^2+C_2)}{C_2}\right)} \tag{3.2.11}$$

Eliminating all the terms that divide out and simplifying allows for a final equation that is a ratio of maximum power in series divided by maximum power of individual modules added together in terms of ratios of Seebeck coefficient (3.0.8) and electrical resistance (3.0.9). The series power ratio is seen in equation (3.2.12).

$$\frac{P_{s,max}}{P_{TE1}+P_{TE2}} = \frac{C_2 (1+C_1)^2}{(C_2+C_1^2)(1+C_2)} \tag{3.2.12}$$

3.3 Parallel

When thermoelectric modules are connected in parallel, they operate with voltage difference being the same across modules but currents differing. Establishing that the voltage is the same across modules and that there are two modules, governing equation (3.1.2) can be written as

$$I_1 = \frac{N\alpha_1 (T_h - T_c)}{NR_{e1}} - \frac{V_p}{NR_{e1}} \tag{3.3.1}$$

$$I_2 = \frac{N\alpha_2 (T_h - T_c)}{NR_{e2}} - \frac{V_p}{NR_{e2}} \tag{3.3.2}$$

for each module. Adding (3.3.1) and (3.3.2) together provides

$$I_1 + I_2 = \Delta T \left(\frac{N\alpha_1}{NR_{e1}} + \frac{N\alpha_2}{NR_{e2}}\right) - \left(\frac{1}{NR_{e1}} + \frac{1}{NR_{e2}}\right) V_p \tag{3.3.3}.$$

To simplify this equation

$$I_1 + I_2 = I_p \tag{3.3.4}$$

so

$$I_p = \Delta T \left(\frac{N\alpha_1}{NR_{e1}} + \frac{N\alpha_2}{NR_{e2}}\right) - \left(\frac{1}{NR_{e1}} + \frac{1}{NR_{e2}}\right) V_p \tag{3.3.5}.$$

Rearranging (3.3.5) with voltage as a function of current for parallel modules gives

$$V_p = \Delta T \left(\frac{N\alpha_1}{NR_{e1}} + \frac{N\alpha_2}{NR_{e2}}\right)\left(\frac{1}{NR_{e1}} + \frac{1}{NR_{e2}}\right)^{-1} - \left(\frac{1}{NR_{e1}} + \frac{1}{NR_{e2}}\right)^{-1} I_p \quad (3.3.6).$$

Maximum power current can be derived by finding short circuit voltage of equation (3.3.5) and dividing by two. Maximum power voltage can be determined by finding open circuit voltage of equation (3.3.6) and dividing by two.

$$I_{p,max} = \frac{I_{sc}}{2} = \left(\frac{\Delta T}{2}\right)\left(\frac{N\alpha_1}{NR_{e1}} + \frac{N\alpha_2}{NR_{e2}}\right) \quad (3.3.7)$$

$$V_{p,max} = \frac{V_{oc}}{2} = \left(\frac{\Delta T}{2}\right)\left(\frac{N\alpha_1}{NR_{e1}} + \frac{N\alpha_2}{NR_{e2}}\right)\left(\frac{1}{NR_{e1}} + \frac{1}{NR_{e2}}\right)^{-1} \quad (3.3.8)$$

Maximum power is equal to the maximum power current multiplied by the maximum power voltage for modules linked electrically in parallel.

$$P_{p,max} = I_{p,max} \times V_{p,max} = \frac{(\Delta T)^2}{4}\left(\frac{N\alpha_1}{NR_{e1}} + \frac{N\alpha_2}{NR_{e2}}\right)^2\left(\frac{1}{NR_{e1}} + \frac{1}{NR_{e2}}\right)^{-1} \quad (3.3.9)$$

Substituting ratios for the Seebeck coefficient (3.0.8) and electrical resistance (3.0.9) yields

$$P_{p,max} = \frac{(\Delta T)^2}{4}\left(\frac{N\alpha_1}{NR_{e1}} + \frac{C_1 N\alpha_1}{C_2 NR_{e1}}\right)^2\left(\frac{1}{NR_{e1}} + \frac{1}{C_2 NR_{e1}}\right)^{-1} \quad (3.3.10).$$

Steps for simplifying equation (3.3.10) can be seen in equations (3.3.11) through (3.3.13).

$$P_{p,max} = \frac{(\Delta T)^2}{4}\left(\frac{C_1 N\alpha_1 + C_2 N\alpha_1}{C_2 NR_{e1}}\right)^2\left(\frac{C_2+1}{C_2 NR_{e1}}\right)^{-1} \quad (3.3.11)$$

$$P_{p,max} = \frac{(\Delta T)^2}{4}\left(\frac{C_1+C_2}{C_2}\right)^2\left(\frac{C_2 NR_{e1}}{C_2+1}\right)\left(\frac{N\alpha_1}{NR_{e1}}\right)^2 \quad (3.3.12)$$

$$P_{p,max} = \frac{(\Delta T)^2}{4}\left(\frac{C_1+C_2}{C_2}\right)^2\left(\frac{C_2}{C_2+1}\right)\frac{(N\alpha_1)^2}{NR_{e1}} \quad (3.3.13)$$

With a final equation of

$$P_{p,max} = \frac{(\Delta T)^2 (N\alpha_1)^2}{4(NR_{e1})}\frac{(C_1+C_2)^2}{C_2(C_2+1)} \quad (3.3.14).$$

To determine the difference between maximum power of two modules in series and adding the same two modules maximum powers together, another ratio is set up. Dividing equation (3.3.14) by equation (3.0.10) creates a ratio that displays mismatch.

$$\frac{P_{p,max}}{P_{TE1}+P_{TE2}} = \frac{\frac{(\Delta T)^2 (N\alpha_1)^2 (C_1+C_2)^2}{4(NR_{e1})\quad C_2(C_2+1)}}{\left(\frac{(N\alpha_1)^2(\Delta T)^2}{4(NR_{e1})}\right)\left(\frac{(C_1^2+C_2)}{C_2}\right)} \quad (3.3.15)$$

Eliminating all the terms that divide out and simplifying allows for a final equation that is a ratio of maximum power in parallel divided by maximum power of individual modules added together in terms of ratios of Seebeck coefficient (3.0.8) and electrical resistance (3.0.9). The parallel power ratio is seen in equation (3.3.16).

$$\frac{P_{p,max}}{P_{TE1}+P_{TE2}} = \frac{(C_1+C_2)^2}{(C_2+C_1^2)(1+C_2)} \tag{3.3.16}$$

3.4 Module Mismatch Experimentation

3.4.1 Thermoelectric Module Test Stand

Experimental data was taken to test the proposed relationships, equations (3.2.12) and (3.3.16), for thermoelectric modules in series and parallel. All of the tests for verification were performed on the thermoelectric module test stand in RIT's SEL. The test stand includes an insulated heating block, cold plate, mechanical system for secure placement of the TEM and thermal isolation, and electrical devices for measurement at different loads as shown in Figure 3.3. The heating block is made of copper to ensure even heat distribution from the 1300 W heater inside the block. All of the sides of the heating block are insulated except the bottom surface which makes contact with the thermoelectric modules. The heater is capable of maintaining a constant, uniform hot side temperature of 500°C. Multiple temperatures are taken near the TEM to obtain a precise measurement of the temperature difference across the module. Electrical resistance is varied within the circuit to gain voltage/current data for characterization of the module(s). Extensive testing and characterization of the test stand was done by Sandoz-Rosado and Stevens [5].

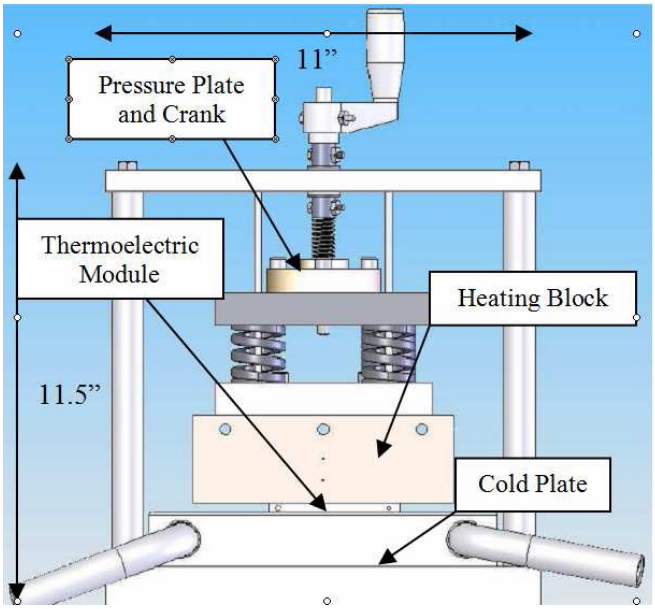


Figure 3.3: Thermoelectric Module Test Stand [5].

3.4.2 Testing

For these tests, hot side temperature was kept at 200°C and cool side temperature at 25°C. A layer of insulation was placed between the heating block and cold plate surrounding the thermoelectric(s) being tested. In actuality, the cool side temperature of the module was approximately 33°C due to the heat absorbed from the heating block. Pressure applied from the mechanical crank system was 690 kPa holding the TEM in place between the heating block and cold plate which maintained excellent thermal contact. An original set of tests were run to characterize five Laird Technologies *HT8,12,F2,4040,TA* modules [35]: 1A, 16A, 19A, 34A, & 1B. The two modules that were the most dissimilar with regards to Seebeck coefficient and internal electrical resistance were tested on the stand at the same time. Modules 1B and 19A were tested individually and then in electrical series. Due to their properties being too similar, analysis of the data didn't allow mismatch to be distinctly observed. Several more tests were conducted to test the theory, one of which was linking a Laird module and a Taihuaxing *TEP1-1264-1.5* module [36] together in electrical series. Properties of the Taihuaxing modules were closer to the Laird modules than expected so this test provided no further answers. A new module configuration was proposed to allow an in depth look at each of the important parameters being observed.

Four modules were placed on the test stand at once: Laird 1A, 19A, 34A, and 16A. The heater was set to 200°C for the hot side temperature while the cool side temperature was set to 25°C. A pressure of 345 kPa was applied to the set of four modules that were held together in a square shape. Each module was tested individually after steady state was reached meaning one module

was active while the other three were passive. The results for the determined parameters can be seen in Table 3.1.

Table 3.1: Individual TEM Parameters.

Module	$N\alpha$	*NRe*	*Rte/N*	P_{max}
	[V/K]	[Ω]	[K/W]	[W]
1A	0.0429	2.860	0.279	4.26
19A	0.0429	5.219	0.319	2.35
16A	0.0438	3.131	0.286	4.06
34A	0.0436	3.945	0.300	3.20

Modules 1A and 19A were then connected in series while modules 16A and 34A were connected in parallel. This was done to "create" two modules with noticeably different parameters. The parameters for the "created" modules can be seen in Table 3.2.

Table 3.2: Combined Module Parameters.

Module	$N\alpha$	*NRe*	*Rte/N*	P_{max}
	[V/K]	[Ω]	[K/W]	[W]
1A&19A	0.0849	7.759	0.282	6.17
16A&34A	0.0437	1.776	0.296	7.16

Several tests were conducted with all four modules connected in various configurations. Using the 1A & 19A series "module" and the 16A & 34A parallel "module", a test was run with them in series and a test with them in parallel. Because of the greater difference in parameters, the effects of mismatch can be observed. Two other tests were run where all four modules were in series and then all four modules were in parallel. The parameters determined from these four configurations can be seen in Table 3.3.

Table 3.3: Parameters from Tests with Four Modules in Different Configurations.

Test	$N\alpha$	*NRe*	*Rte/N*	P_{max}
	[V/K]	[Ω]	[K/W]	[W]
(1A&19A)&(16A&34A) in series	0.1287	9.286	0.281	11.82
(1A&19A)&(16A&34A) in parallel	0.0514	1.458	0.279	12.00
1A&19A&16A&34A in series	0.1701	14.429	0.306	13.22
1A&19A&16A&34A in parallel	0.0434	0.930	0.285	13.46

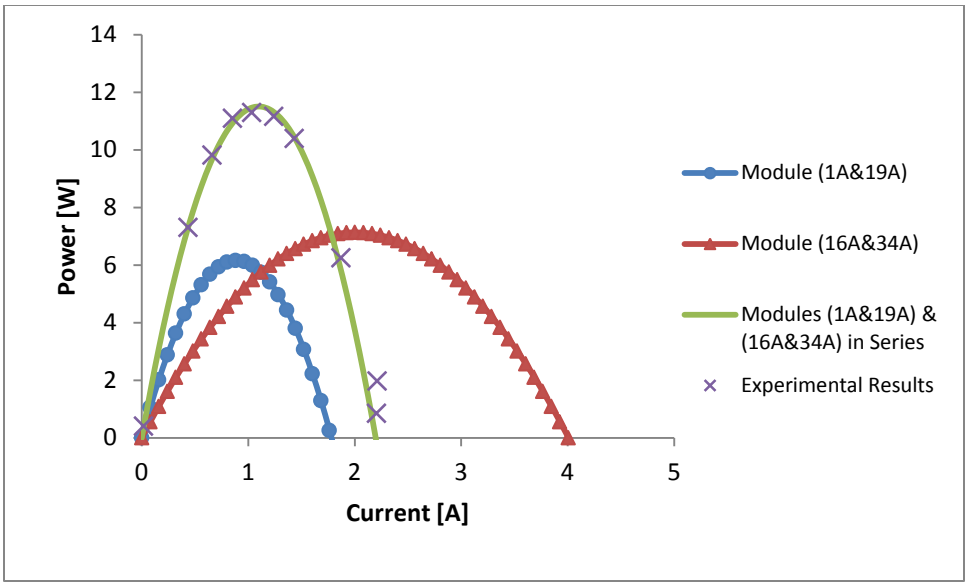


Figure 3.4: Theoretical predictions and experimental results for (1A&19A) and (16A&34A) in series.

Figure 3.4 and Figure 3.5 show experimental results and theoretical calculations for the two tests with the 1A & 19A series "module" and the 16A & 34A parallel "module" in series and parallel. The experimental results match closely with the theoretical predictions in both cases.

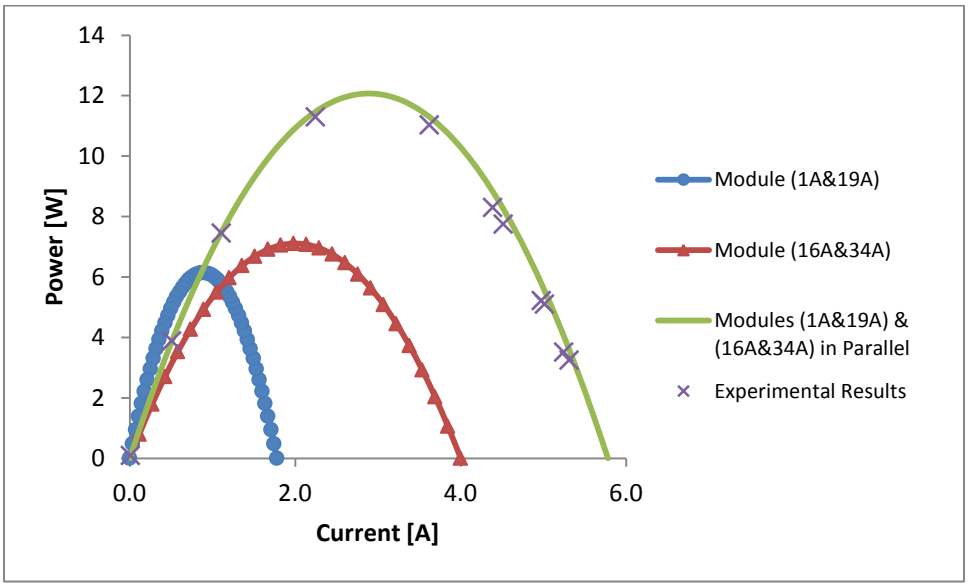


Figure 3.5: Theoretical predictions and experimental results for (1A&19A) and (16A&34A) in parallel.

Each of the modules in their various configurations provided different maximum values for power in watts as seen in Table 3.4. These values are used to compare theoretical predictions to the experimental data obtained to provide answers to the closeness in the predictions of the ratios of power determined earlier.

Table 3.4: Maximum Power for Each of the Module Configurations.

Module Configuration	P_{max}
	[W]
1A	4.26
19A	2.35
16A	4.06
34A	3.20
19A&1A in series	6.17
16A&34A in parallel	7.16
(1A&19A)&(16A&34A) in series	11.82
(1A&19A)&(16A&34A) in parallel	12.00
1A&19A&16A&34A in series	13.22
1A&19A&16A&34A in parallel	13.46

Using data collected from all of the experiments, comparison of experimental and theoretical can be made as seen in Table 3.5.

Table 3.5: Comparison of Experimental to Theoretical Ratios.

Test	$P_{max,exp}$ [W]	$\frac{P_{max,exp}}{P_{TE1,exp}+P_{TE2,exp}}$	$\frac{P_{max}}{P_{TE1}+P_{TE2}}$	*% Difference*
1A&19A in series	6.17	0.933	0.915	1.93%
16A&34A in parallel	7.16	0.985	1.000	1.50%
(1A&19A)&(16A&34A) in series	11.82	0.887	0.865	2.48%
(1A&19A)&(16A&34A) in parallel	12.00	0.900	0.911	1.21%

The first column is the maximum power during the experimental test. Column two takes the ratio of the maximum power from the experiment and divides it by the maximum power of the individual modules added together which can be found in Table 3.4. Column three uses the theoretical equations (3.2.12) and (3.3.16) depending on whether they are in series or parallel, to predict the ratio of maximum power for each case dependent on the parameters in Table 3.1 and Table 3.2. Column four shows the percent difference between the experimental and theoretical ratios.

For each mismatch test, percent differences were within 2.5% showing that the models for mismatch in series and parallel configurations (equations (3.2.12) and (3.3.16), respectively), are a strong predictor of actual power output. Most of this error is a result of variations in the measurement equipment because all data was taken without adjusting the testing configuration. All tests were run under the same operating conditions in the same day.

The ratios of powers, shown in the second and third columns of Table 3.5, display that module mismatch accounts for approximately 10% or less of power losses when module parameters are

significantly different. Slightly higher percentages of lost power can be credited to module mismatch when parameters start to differ drastically. This is dependent on the values of the ratios of the parameters from equations (3.0.8) and (3.0.9).

3.5 Summary

Based on the model for thermoelectric modules in series from equation (3.2.12), it is actually possible for two thermoelectric modules to vary by ratios significantly greater or less than unity and mismatch be a non-issue as long as the ratios are the same numerical value. When C_1 equals C_2, equation (3.5.1) is developed with C in place of the two ratios and results in the power ratio equal to one.

$$\frac{P_{s,max}}{P_{TE1}+P_{TE2}} = \frac{C(1+C)^2}{(C+C^2)(1+C)} = 1 \tag{3.5.1}$$

This observation does not hold true for thermoelectric modules in parallel. Putting C in place of C_1 and C_2 in equation (3.3.16) yields

$$\frac{P_{p,max}}{P_{TE1}+P_{TE2}} = \frac{(C+C)^2}{(C+C^2)(1+C)} = \frac{4C}{(C+1)^2} \tag{3.5.2}$$

This ratio provides evidence that there will be mismatch regardless of the similarity of the values of the parameter ratios.

Another interesting mathematical property of equations (3.2.12) and (3.3.16) to observe is when C_1 is equal to one. For series ratio equation (3.2.12), replacing C_1 with one becomes equation (3.5.3).

$$\frac{P_{s,max}}{P_{TE1}+P_{TE2}} = \frac{C_2(1+1)^2}{(C_2+1^2)(1+C_2)} = \frac{4C_2}{(C_2+1)^2} \tag{3.5.3}$$

This equation closely resembles equation (3.5.2) which was the parallel ratio when C_1 equals C_2.

When C_1 equals one in equation (3.3.16), the following phenomena occurs

$$\frac{P_{p,max}}{P_{TE1}+P_{TE2}} = \frac{(1+C_2)^2}{(C_2+1)(1+C_2)} = 1 \tag{3.5.4}$$

The ratio equals one meaning that as long as the Seebeck Coefficient for each module is the same, module mismatch will not affect the performance of the thermoelectric modules when connected in parallel configuration.

A final case to observe is when C_2 is set to one meaning that the electrical resistance for each thermoelectric module is equal. Using equations (3.2.12) and (3.3.16), the result is the same for both series and parallel configurations and can be seen in equation (3.5.5).

$$\frac{P_{max}}{P_{TE1}+P_{TE2}} = \frac{(C_1+1)^2}{2(1+C_1^2)} \tag{3.5.5}$$

The most interesting concept from this relationship is that as long as the electrical resistances are the same, the power ratio can never be less than 0.5. This holds true for both thermoelectric modules in series and parallel.

The module mismatch problem was first identified in Smith's thesis work [9]. He used a similar approach to that shown in equation (3.2.12) but saw large discrepancies in his results. These great differences in predicted and experimental results were most likely the result of repeatability issues with the test stand and equipment, the modules, and their setup. The reported tests above were run as a set of tests and all performed in a single setup with time in-between for the modules to reach steady state. This removed the uncertainty that may have resulted from the heating and cooling of the system, thermal paste, and modules.

Overall, mismatch is not a profound issue, even when the values of the parameters differ significantly. Besides the special cases when C_1 equals C_2 in series and C_1 equals one in parallel, parameter ratios that stray from unity cause thermoelectric modules to not perform as expected through simple superposition of individual modules' IV curves but the effect is small. Power ratios equation (3.2.12) and equation (3.3.16) provide accurate prediction for the issue of module mismatch which will account for any power losses due to mismatch, even those that are insignificant for modules that have close parameters.

4 Heat Spreading in Three Dimensions

Heat spreading in three dimensions is currently not considered in many system models for thermoelectric heat exchangers. Accounting for this phenomenon will provide a better understanding of differences between model predictions and experimental results. Understanding discrepancies in predicted power compared to actual power recovered when heat spreading occurs in extended surfaces that have different aspect ratios than the thermoelectric modules is of importance. This does not permit the assumption of one dimensional heat transfer, but rather requires modeling for the three dimensional heat spreading. System tests and finite element analysis have shown that prediction of power generation is inaccurate when the ratios of surface areas of the bases of extended surfaces and thermoelectric devices differ. An analytical approach is needed to compensate for this phenomenon as opposed to an experimental or numerical approach for reasons discussed in Section 2.3. A quick and accurate solution is desirable so that it can be referenced in an improved steady state 1-D model being used by the optimization routine in TEPSS. This requires a calculated effective thermal resistance as a result of heat spreading in three dimensions due to the non-unity aspect ratios of the thermoelectric modules to their extended surfaces. The use of an existing model found in the literature with an adaptation or modification provides this effective thermal resistance and analytical solution. Experimental data is gathered, comparing a set of fins to the 3-D analytical solution for verification.

4.1 Background & Theory

When a heat source is mated to a surface, conduction will occur. When this heat source is not of the same area as the mated surface, heat spreading will occur in three dimensions as illustrated in Figure 4.1. This concept can be extended to thermoelectric modules placed on finned heat sinks.

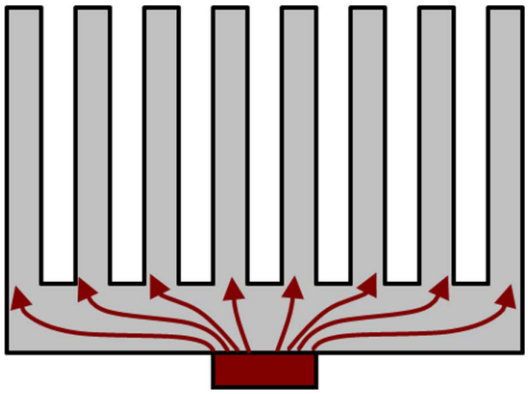


Figure 4.1: Heat flux path that occurs due to heat spreading in three dimensions from a heat source to finned heat sinks [1].

The current thermoelectric system model developed by Smith [9] uses finite element analysis to predict the heat conduction through the fins and apply it to the solution. Prior to the

implementation of the FEA, only 1D conduction was assumed through the testing power unit and the fins. This didn't provide accurate results for Smith; therefore, a 3D heat spreading conduction was added to the model. The finite element analysis is representative of only one specific fin geometry. The existing model currently only predicts heat spreading for one type of fin arrangement which allows for no variability in design of other systems. Calculation of heat spreading effects for other fin geometries would require many FEA solutions to developed, whereas a universal solver based on analytical methods would remove the need for FEA altogether. Three-dimensional conduction may then be predicted for any fin geometry and aspect ratio of thermoelectric devices to finned plates.

Thermal spreading of heat from the thermoelectric through the fins can be evaluated using a specific model that will take input size parameters, non-dimensionalize them, and provide an effective thermal resistance. This effective resistance includes the three-dimensional spreading resistance and the one-dimensional resistance, in lieu of the FEA 3D spreading resistance. Using concepts and mathematical procedures developed by Ellison [14], thermal spreading resistance is calculated for exact analytical solutions. This will allow for variation in the shape, size, and number of fins in the heat sink design. The end result will be a greater versatility in the modeling capabilities of the thermoelectric power unit component model.

4.2 Model

The model developed by Ellison analytically determines an exact effective thermal resistance between a rectangular heat source coupled to larger rectangular plate with convection on the opposite side [14]. The model is capable of separating the effective resistance into components that represent the one-dimensional resistance and the spreading resistance. The model solves problems that are represented by the geometry in Figure 4.2. A plate is defined by dimensions a, b, and t in coordinates x, y, and z, respectively. A uniform heat source is defined by dimensions Δx and Δy in the x and y coordinates, respectively. The heat source has a magnitude of Q located on the $z=0$ plane and is shown with a thickness to clearly represent its location in the xz plane view. The actual heat source is taken to have a thickness of zero when solving the problem and is centered on the plate as can be seen in the xy plane view in Figure 4.2.

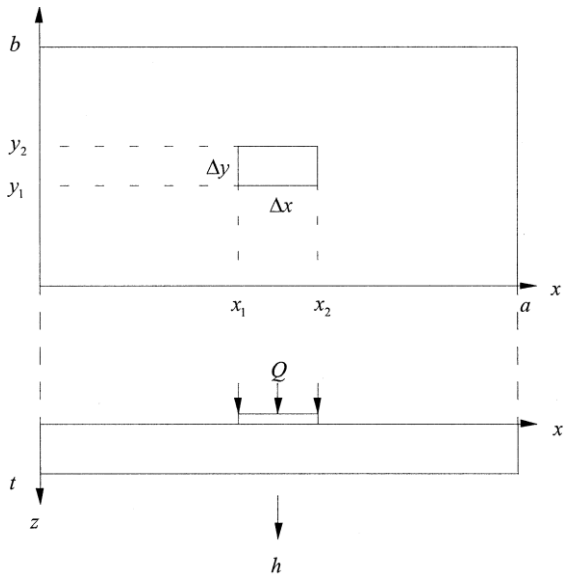


Figure 4.2: Geometry for the thermal spreading problem viewed from the xy plane and the xz plane [14].

The steady state heat conduction partial differential equation represents the governing equation for this problem as can be seen in equation (4.2.1). Directional conductivity is constant in all directions, i.e. conduction coefficient in the x-direction equals conduction coefficient in the y-direction which equals conduction coefficient in the z-direction.

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} = -\frac{Q_v}{k} \tag{4.2.1}$$

The boundary conditions for the proposed geometry in Figure 4.2 are shown in (4.2.2)-(4.2.5). The four boundary conditions in (4.2.2) and (4.2.3) define the plate edges as adiabatic. The boundary condition in (4.2.4) creates an adiabatic surface on the surface on which that the source is located at $z=0$. The adiabatic condition applies everywhere on that plane except where the source is defined by Δx and Δy. Boundary condition (4.2.5) represents the surface with convective heat transfer across it at $z=t$. Convective coefficient, h, is assumed to be uniform across the entire surface. The ambient temperature (T) is set to zero so that all calculated temperatures are considered excess temperatures above ambient. The experimental setup uses insulation to cover the areas that are considered adiabatic. This justifies this assumption and gives it physical world applicability.

$$k\frac{\partial T}{\partial x} = 0 \; at \; x = 0, a \tag{4.2.2}$$

$$k\frac{\partial T}{\partial y} = 0 \; at \; y = 0, b \tag{4.2.3}$$

$$k\frac{\partial T}{\partial z} = 0 \; at \; z = 0 \tag{4.2.4}$$

$$k\frac{\partial T}{\partial z} = -hT \; at \; z = t \tag{4.2.5}$$

Equation (4.2.6) displays the uniform heat source density Q_V. It consists of double Fourier cosine series in the *xy*-plane and *z*-dependent Fourier coefficients $\varphi_{lm}(z)$. Equation (4.2.7) displays the temperature solution in a similar format to the heat source.

$$Q_v(x,y,z) = \sum_{l=0}^{\infty}\sum_{m=0}^{\infty}\varepsilon_l\varepsilon_m\phi_{lm}(z)\cos\left(\frac{l\pi x}{a}\right)\cos\left(\frac{m\pi y}{b}\right) \tag{4.2.6}$$

$$T(x,y,z) = \sum_{l=0}^{\infty}\sum_{m=0}^{\infty}\varepsilon_l\varepsilon_m\psi_{lm}(z)\cos\left(\frac{l\pi x}{a}\right)\cos\left(\frac{m\pi y}{b}\right) \tag{4.2.7}$$

The heat flux at the source is assumed to be uniform at the *z=0* plate surface using a Dirac delta function. The Fourier series coefficients are determined and then the Fourier expansion results are applied to the steady state heat conduction equation in (4.2.1). This converts the 3D problem into a 1D solvable problem. Green's functions are used to complete the answer resulting in *T(x, y, z)* shown in equations (4.2.8)-(4.2.10).

$$T(x,y,z) = T_U + T_{Sp} \tag{4.2.8}$$

$$T_U = \frac{\Psi_{00}}{4} \tag{4.2.9}$$

$$T_{Sp} = \sum_{l=1}^{\infty}\left(\frac{1}{2}\right)\psi_{l0}\cos\left(\frac{l\pi x}{a}\right) + \sum_{m=1}^{\infty}\left(\frac{1}{2}\right)\psi_{0m}\cos\left(\frac{m\pi y}{b}\right) + \sum_{l=1}^{\infty}\sum_{m=1}^{\infty}\psi_{lm}\cos\left(\frac{l\pi x}{a}\right)\cos\left(\frac{m\pi y}{b}\right) \tag{4.2.10}$$

The next step to finding the effective thermal resistance is to take the temperature function *T(x, y, z)* and divide by the source dissipation *Q* resulting in a resistance. The coordinates for centering the source are required to be inserted into the equation for the resistance. The terms in the equation are then non-dimensionalized using the variables $\alpha = \Delta x/a$, $\beta = \Delta y/a$, $\rho = a/b$, $\tau = t/a$, and $Biot\cdot\tau = (ha/k)\cdot(t/a) = ht/k$. Equation (4.2.11) shows the effective thermal resistance equal to its one-dimensional component and its spreading component. Equation (4.2.12) is the same relationship using the non-dimensionalized variables.

$$R = R_U + R_{Sp} \tag{4.2.11}$$

$$\frac{\Psi}{k\sqrt{\Delta x\Delta y}} = \frac{\Psi_U}{k\sqrt{\Delta x\Delta y}} + \frac{\Psi_{Sp}}{k\sqrt{\Delta x\Delta y}} \tag{4.2.12}$$

The effective thermal resistance from the defined source requires the average temperature of the source using the area integration of the temperature function for that region as shown in equation (4.2.13).

$$\bar{T} = \frac{1}{\Delta x\Delta y}\int_{x1}^{x2}\int_{y1}^{y2}T(x,y,z=0)dxdy \tag{4.2.13}$$

This transforms equations (4.2.11) and (4.2.12) to averaged results represented by equations (4.2.14) and (4.2.15) shown below.

$$\bar{R} = R_U + \bar{R}_{Sp} \tag{4.2.14}$$

$$\frac{\bar{\Psi}}{k\sqrt{\Delta x \Delta y}} = \frac{\Psi_U}{k\sqrt{\Delta x \Delta y}} + \frac{\bar{\Psi}_{Sp}}{k\sqrt{\Delta x \Delta y}} \tag{4.2.15}$$

Ψ_U is the dimensionless 1D resistance and shown in (4.2.16). $\bar{\Psi}_{Sp}$ is the dimensionless, source averaged spreading resistance and can be seen in equation (4.2.17).

$$\Psi_U = \rho\sqrt{\alpha\beta\tau}\left(1 + \frac{1}{Biot\cdot\tau}\right) \tag{4.2.16}$$

$$\bar{\Psi}_{Sp} = \left(\frac{\rho}{\pi^3\alpha}\right)\sqrt{\frac{\beta}{\alpha}}\sum_{l=1}^{\infty}\frac{1}{l^3}sin^2(l\pi\alpha)\left[\frac{1+\left(\frac{Biot\cdot\tau}{2l\pi\tau}\right)tanh(2l\pi\tau)}{\left(\frac{Biot\cdot\tau}{2l\pi\tau}\right)+tanh(2l\pi\tau)}\right] +$$

$$\left(\frac{1}{\pi^3\rho^2\beta}\right)\sqrt{\frac{\alpha}{\beta}}\sum_{m=1}^{\infty}\frac{1}{m^3}sin^2(m\pi\beta\rho)\left[\frac{1+\left(\frac{Biot\cdot\tau}{2m\pi\rho\tau}\right)tanh(2m\pi\rho\tau)}{\left(\frac{Biot\cdot\tau}{2m\pi\rho\tau}\right)+tanh(2m\pi\rho\tau)}\right] +$$

$$\left(\frac{4}{\pi^4\rho\alpha\beta\sqrt{\alpha\beta}}\right)\sum_{l=1}^{\infty}\sum_{m=1}^{\infty}\frac{1}{l^2m^2}sin^2(l\pi\alpha)sin^2(m\pi\beta\rho)\times$$

$$\left\{\frac{1+\left(\frac{Biot\cdot\tau}{2\pi\tau\sqrt{l^2+m^2\rho^2}}\right)tanh\left(2\pi\tau\sqrt{l^2+m^2\rho^2}\right)}{2\pi\sqrt{l^2+m^2\rho^2}\left[\left(\frac{Biot\cdot\tau 2\pi\tau}{\sqrt{l^2+m^2\rho^2}}\right)+tanh\left(2\pi\tau\sqrt{l^2+m^2\rho^2}\right)\right]}\right\} \tag{4.2.17}$$

The results from equations (4.2.16) and (4.2.17) can be added together and re-dimensionalized to provide an effective thermal resistance to be used in the thermoelectric system model. The model is tested by Ellison with promising results for geometries similar to the one in Figure 4.2 [14].

Overall, the model is very effective for the proposed geometry. It operates under some significant approximations and limitations under all cases. These assumptions dictate that the source is centrally located and delivers uniform heat flux, the source plane is adiabatic except for the source region, and the substrate edges are adiabatic. Also, the heat transfer coefficient is uniformly distributed across the plane opposite the source plane. Both the thermal conductivity and the heat transfer coefficient are treated as though they are constant over the range of temperatures.

Another limiting aspect is that this model is not designed for use with finned heat sink geometries like in Figure 4.1 but rather for flat plates like Figure 4.2. For simple analysis of finned heat sinks, Ellison proposes to use an effective heat transfer coefficient calculated by use

of an effective area which accounts for the additional surface area of the fins. The effective heat transfer coefficient is

$$h_e = h\left[\frac{A_f}{(ab)}\right] \quad (4.2.18)$$

where

$$A_f = 2N_f Lb + ab \quad (4.2.19).$$

This proposed extension of the model assumes each of the fins operate at ideal efficiency. The accuracy of this assumption is examined in comparison to numerical solutions provided by ANSYS® in the next section.

4.3 Numerical Solutions

Finite Element Analysis software was used to provide numerical solutions to potential fin geometries of interest. ANSYS® was used for most of the simulations except when COMSOL® was used in early stages of simulations to verify that the model in ANSYS® was accurately configured. Final results were compared between the two programs with good agreement.

When using ANSYS®, the models were created using keypoints and then creating volumes from these keypoints. All volumes were "glued" together and meshed using element type *Solid Brick8node70*. Preferences were set to thermal and the conduction coefficient (*kxx*) was set to 200 W/(m-K) for all volumes of each model. Boundary conditions mimicked those of the Ellison model with adiabatic surfaces around the edges and source plane. The source was defined to have a heat flux of 10^6 W/m^2 for all cases and the source was applied to a single surface. The convection coefficient varied between simulations but was defined for all surfaces comprising the fins. Adiabatic boundary conditions were applied to the front and back face of each fin.

Models were solved and an average temperature of the source face was calculated using a surface averaging function in ANSYS®. The average temperature was divided by the heat flow through the source giving an effective thermal resistance in ANSYS® which could be compared to the effective thermal resistance found from the Ellison model programmed in MATLAB®.

Before adding fins to the FEA model, a check was performed comparing the ANSYS® result to the model. The geometry used was like Figure 4.2 with a = 0.1 m, b = 0.1 m, t = 0.005 m, Δx = 0.01 m, Δy = 0.01 m, $q_o = 10^6$ W/m^2, and h = 10 W/(m^2-K). The analytical solution from the Ellison model was R = 10.35 K/W while the numerical solution from ANSYS® was R = 10.35 K/W. This provided confidence in the analytical model and the FEA, before conducting more sophisticated analysis with finned heat exchangers.

Initially, two types of geometries were proposed so that they were similar in size but fins varied in direction. The effective fin area was the same for both geometries in order to ensure that the model could be closely examined. Figure 4.3 shows type A which had thirty fins that extended for a shorter distance while Figure 4.4 shows type B which had three fins that extended through a longer distance. The three longer fins are equivalent in effective area to the thirty shorter fins. For both of these models, the MATLAB® code would predict the same value for the effective thermal resistance. However, ANSYS® provided a different result for each case. That raised questions about heat spreading along the fins, which is assumed to be negligible when applying the Ellison model. The results from ANSYS® did not agree with the Ellison model. Also coming into question was the validity of equation (4.2.19) which did not take into account the thickness of the fins or their efficiency. A final concern was that spreading between fins was occurring which is not accounted for in the analytical model.

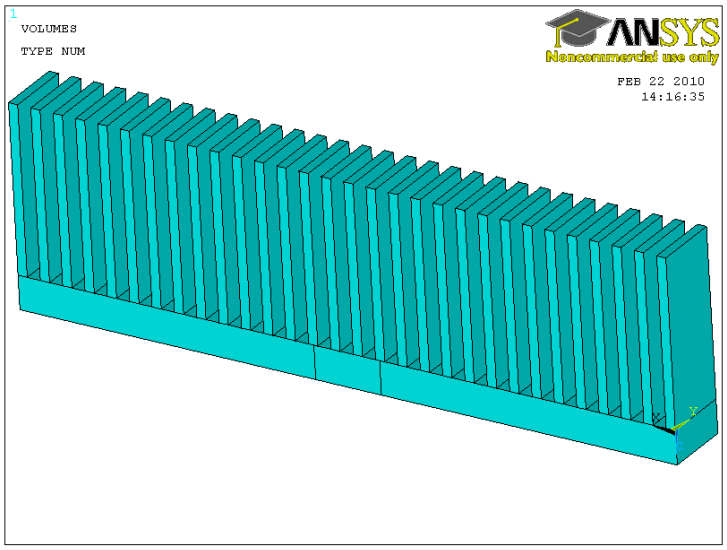


Figure 4.3: Model Type A with More Fins that Extend Through a Shorter Distance.

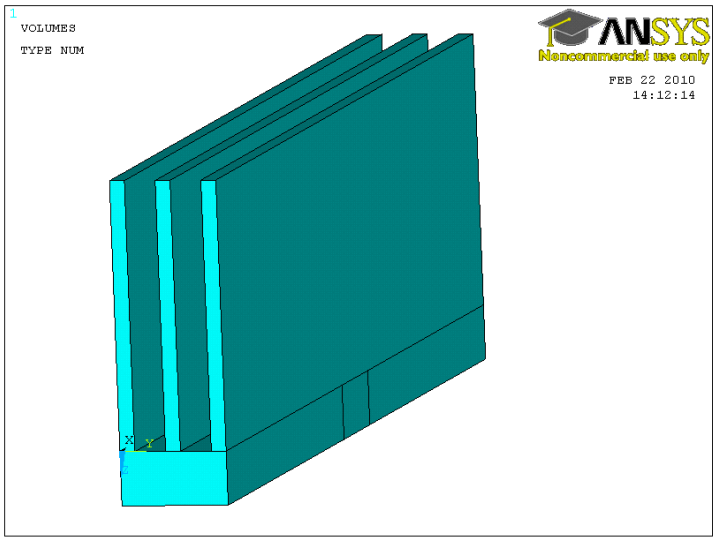


Figure 4.4: Model Type B with Fewer Fins that Extend Through a Longer Distance.

Another type of geometry was used to eliminate some of the effects of spreading along the finned heat sinks. The sink to source ratio was 100% and can be seen in Figure 4.5. This was used to eliminate the 3D effects from the spreading phenomena.

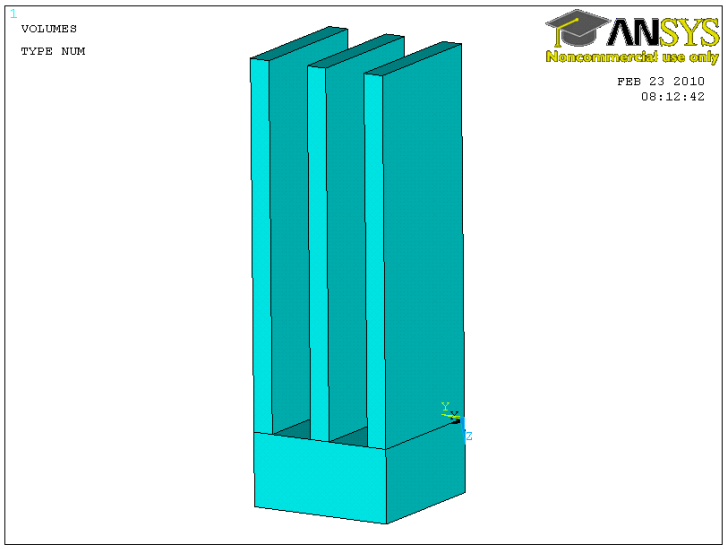


Figure 4.5: Model with Sink to Source Ratio of 100%.

To analyze the effect of adding fin efficiency to the model, the following equations were used to make necessary calculations. These equations are standard for extended surfaces and come from Incropera et al. [37]. Figure 4.6 displays the representation of each of the variables.

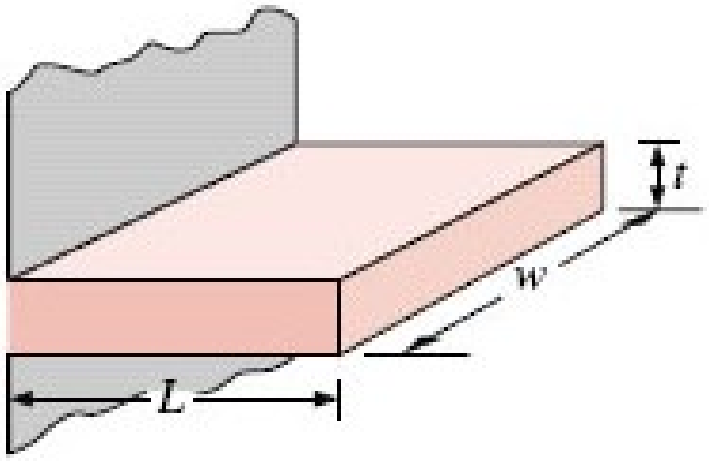


Figure 4.6: Orientation and Variables of a Rectangular Extended Surface [37].

The equation for efficiency of a single rectangular fin is

$$\eta_f = \frac{tanh(mL_c)}{mL_c} \tag{4.3.1}$$

where

$$L_c = L + \frac{t}{2} \tag{4.3.2}$$

and

$$m = \sqrt{\frac{hP}{kA_c}} \tag{4.3.3}$$

where

$$P = 2w + 2t \tag{4.3.4}$$

and

$$A_c = wt \tag{4.3.5}.$$

The addition of the fin efficiency will change the effective thermal coefficient, h_e. Also, taking into account the thickness of each fin, t_f, will change the effective fin area, A_f. The new equation for the effective area is (4.3.6).

$$A_f = \left(a - N_f t_f\right)b + 2\eta_f L_c b \tag{4.3.6}$$

These changes to this equation directly affect the value for effective thermal coefficient obtained using equation (4.2.18). The affect this change has can be seen throughout the results listed in tables below.

Table 4.1 displays seven cases that have a low convective coefficient that is close to a value of natural convection. Cases A1 and A3 resemble the geometry in Figure 4.3 while Cases A2 and A4 resemble the geometry in Figure 4.4. Cases A1 and A2 have the same effective area as well as Cases A3 and A4. In each comparison, both the basic model and the model using efficiency predict resistance values very close to each other. However, the ANSYS® predicts values that differ from the Ellison model. Cases A5-A7 uses the geometry in Figure 4.5 which has a sink to source ratio of 100%. This eliminates the effects of heat spreading along the fins. Comparison of A5, A6, and A7 allow us to observe the effect of fin efficiency on thermal resistance. Case A7 has considerably lower fin efficiency than A5 and A6 but it can be seen that the percent difference between the model using the efficiency and ANSYS® is lower in each case than when the basic model is used.

Table 4.1: Test Cases Set A with Common Properties of t = 0.005 m, Δx = 0.01 m, Δy = 0.01 m, and h = 10 W/(m^2-K).

Case	*Case Properties*	*R* *Basic Model*	*R* *Model Using η_f*	η_f	*R* *ANSYS*	*% Difference Basic Model to ANSYS*	*% Difference Model Using η_f to ANSYS*
A1	a=0.1 b=0.01 L_f=0.05 N_f=30 t_f=0.0014	3.9544	4.1529	0.9415	4.1294	4.24%	-0.57%
A2	a=0.01 b=0.1 L_f=0.05 N_f=3 t_f=0.0014	3.9509	4.1493	0.9415	3.8649	-2.23%	-7.36%
A3	a=0.1 b=0.01 L_f=0.025 N_f=30 t_f=0.0014	6.9957	7.0956	0.9837	7.3271	4.52%	3.16%
A4	a=0.01 b=0.1 L_f=0.025 N_f=3 t_f=0.0014	6.9921	7.0921	0.9837	6.8047	-2.75%	-4.22%
A5	a=0.01 b=0.01 L_f=0.025 N_f=3 t_f=0.0014	62.7500	63.7464	0.9837	66.1480	5.14%	3.63%
A6	a=0.01 b=0.01 L_f=0.05 N_f=3 t_f=0.0014	32.5081	34.4727	0.9415	34.4480	5.63%	-0.07%
A7	a=0.01 b=0.01 L_f=0.05 N_f=3 t_f=0.00025	32.5081	42.3047	0.7599	42.4080	23.34%	0.24%

The case tests in Table 4.2 mimic the cases in Table 4.1 but have a higher convective coefficient. The new value for convection is 200 W/(m^2-K) and is used for the remaining simulations. This value is closer to the value that can be expected in actual system performance evaluations. Fin efficiency is considerably lower in every case because of the higher convection coefficient which allows observation of the affect of fin efficiency. Cases B1 and B2 have the same effective area and so do Cases B3 and B4. When observing the percent difference for the model using fin efficiency, it appears as though B1 and B3 are very accurate but B2 and B4 are not. In both B2 and B4, the fins extend over a longer distance causing heat spreading along the fins. In Cases B5-B7, the fins do not extend over a longer distance because the sink to source ratio is changed to 1. In each of these cases, the results for the effective thermal resistance using the basic model are drastically different than the ANSYS® results. However, the results are very close when incorporating fin efficiency into the model. This provides evidence that fin efficiency is a necessary factor in accurately determining the thermal resistance of a fin array.

Table 4.2: Test Cases Set B with Common Properties of t = 0.005 m, Δx = 0.01 m, Δy = 0.01 m, and h = 200 W/(m^2-K).

Case	*Case Properties*	*R* *Basic Model*	*R* *Model Using η_f*	η_f	*R* *ANSYS*	*% Difference* *Basic Model to ANSYS*	*% Difference* *Model Using η_f to ANSYS*
B1	a=0.1 b=0.01 L_f=0.05 N_f=30 t_f=0.0014	0.6103	0.8603	0.4940	0.8472	27.96%	-1.55%
B2	a=0.01 b=0.1 L_f=0.05 N_f=3 t_f=0.0014	0.6067	0.8568	0.4940	0.7451	18.58%	-14.99%
B3	a=0.1 b=0.01 L_f=0.025 N_f=30 t_f=0.0014	0.8493	0.9751	0.7622	0.9664	12.12%	-0.90%
B4	a=0.01 b=0.1 L_f=0.025 N_f=3 t_f=0.0014	0.8457	0.9716	0.7622	0.8379	-0.93%	-15.96%
B5	a=0.01 b=0.01 L_f=0.025 N_f=3 t_f=0.0014	3.3750	4.3041	0.7622	4.2836	21.21%	-0.48%
B6	a=0.01 b=0.01 L_f=0.05 N_f=3 t_f=0.0014	1.8629	3.4537	0.4940	3.4442	45.91%	-0.28%
B7	a=0.01 b=0.01 L_f=0.05 N_f=3 t_f=0.00025	1.8629	6.8161	0.2224	6.9230	73.09%	1.54%

The simulations run in Table 4.3 look into the effects of heat spreading along the fins as the fins extend further from the heat source. Case C1 serves as a base case and has a sink to source ratio of 1. The comparison of the model using fin efficiency to ANSYS® results for effective thermal resistance provides evidence that heat spreading is not an issue when the ratio is 1. As the ratio increases, the percentage difference increases for comparison of the model with fin efficiencies incorporated. As this percentage difference worsens, the % difference for the basic model gets closer to zero. This happens because the inaccuracy that results from not incorporating fin efficiencies compensates for the inaccuracy caused by the heat spreading along the fins causing them to zero out. It is still safe to have confidence in the model using fin efficiency because Cases C3 through C6 are unrealistic scenarios. Case C2 is more of a realistic limit for possible real world scenarios involving a thermoelectric power unit because modules will occupy much of the available heat transfer area, leaving little space between them and maintaining an aspect ratio close to 100%. This observation is further explored in Table 4.5.

Table 4.3: Test Cases Set C with Common Properties of t = 0.005 m, Δx = 0.01 m, Δy = 0.01 m, t_f = 0.0014 m, L_f = 0.025 m, N_f = 3, and h = 200 W/(m^2-K).

Case	*Case Properties*	*R* *Basic Model*	*R* *Model Using η_f*	η_f	*R* *ANSYS*	*% Difference Basic Model to ANSYS*	*% Difference Model Using η_f to ANSYS*
C1 (B5)	a=0.01 b=0.01	3.3750	4.3041	0.7622	4.2836	21.21%	-0.48%
C2	a=0.01 b=0.03	1.2768	1.5875	0.7622	1.5628	18.30%	-1.58%
C3	a=0.01 b=0.05	0.9645	1.1568	0.7622	1.0968	12.06%	-5.47%
C4	a=0.01 b=0.07	0.8790	1.0277	0.7622	0.9318	5.67%	-10.29%
C5	a=0.01 b=0.09	0.8512	0.9818	0.7622	0.8587	0.87%	-14.33%
C6 (B4)	a=0.01 b=0.1	0.8457	0.9716	0.7622	0.8379	-0.93%	-15.96%

Table 4.4 displays the effects of changing the base thickness of the fin array. Varying the base thickness hasn't been explored in any of the previous simulations. For all prior analysis, the thickness has been t = 0.005 m so in the cases below, it is either double or half that value. Cases D1 through D3 have a sink to source ratio of 1 while Cases D4 through D6 have a ratio of 3. This was determined to be a realistic limit from the simulations in Table 4.3. When comparing the model that uses fin efficiency, it can be seen that the results are very accurate. This allows for the conclusion that differences in base thickness have minimal impact on the accuracy of calculations for effective thermal resistance.

Table 4.4: Test Cases Set D with Common Properties of Δx = 0.01 m, Δy = 0.01 m, t_f = 0.0014 m, L_f = 0.025 m, N_f = 3, and h = 200 W/(m^2-K).

Case	*Case Properties*	*R* *Basic Model*	*R* *Model Using η_f*	η_f	*R* *ANSYS*	*% Difference* *Basic Model to ANSYS*	*% Difference* *Model Using η_f to ANSYS*
D1	a=0.01 b=0.01 t=0.0025	3.2500	4.1791	0.7622	4.1587	21.85%	-0.49%
D2	a=0.01 b=0.01 t=0.005	3.3750	4.3041	0.7622	4.2836	21.21%	-0.48%
D3	a=0.01 b=0.01 t=0.01	3.6250	4.5541	0.7622	4.5340	20.05%	-0.44%
D4	a=0.01 b=0.03 t=0.0025	1.3064	1.6210	0.7622	1.5493	15.68%	-4.63%
D5	a=0.01 b=0.03 t=0.005	1.2768	1.5875	0.7622	1.5628	18.30%	-1.58%
D6	a=0.01 b=0.03 t=0.01	1.3369	1.6467	0.7622	1.6358	18.27%	-0.67%

The twelve simulations in Table 4.5 look further into heat spreading along the fins. It also displays results based on changing the ratio of cross sectional area of the fins to the cross sectional area of the base by changing the thickness of the base. Equations (4.3.7) and (4.3.8) show how the cross sectional area of the fins and base, respectively, are determined.

$$A_f = N_f t_f L_f \tag{4.3.7}$$

$$A_b = ta \tag{4.3.8}$$

The ratio of these areas set up the columns 1x, 2x, and 4x. Rows E1, E2, E3, and E4 refer to the sink to source ratio with values of 1, 1.5, 2, and 3, respectively. This provides incremental increases in the heat spreading along the fins while allowing for observation of cross sectional areas in ratio form. Simulation 1x-E1 provides very accurate results as well as each of the other simulations. To note, the result from MATLAB® uses the model with the addition of fin efficiency. Simulation 4x-E4 has the highest percent difference with *-4.63%* which can be credited primarily to the spreading along the fins.

Table 4.5: Test Cases Set E with Common Properties of Δx = 0.01 m, Δy = 0.01 m, t_f = 0.0014 m, L_f = 0.025 m, N_f = 3, and h = 200 W/(m^2-K).

			1x	*2x*	*4x*
Case	*Description*	*Case Properties*	*t=0.01*	*t=0.005*	*t=0.0025*
			A_f/A_b=1.05	*A_f/A_b=2.1*	*A_f/A_b=4.2*
E1	100% Sink to Source Ratio	a=0.01 b=0.01	ANSYS: R= 4.5340 MATLAB: R= 4.5541 %Difference= -0.44%	ANSYS: R= 4.2836 MATLAB: R= 4.3041 %Difference= -0.48%	ANSYS: R= 4.1587 MATLAB: R= 4.1791 %Difference= -0.49%
E2	150% Sink to Source Ratio	a=0.01 b=0.015	ANSYS: R= 3.0540 MATLAB: R=3.0675 %Difference=-0.44%	ANSYS: R= 2.8874 MATLAB: R= 2.9016 %Difference= -0.49%	ANSYS: R= 2.8070 MATLAB: R= 2.8247 %Difference= -0.63%
E3	200% Sink to Source Ratio	a=0.01 b=0.02	ANSYS: R= 2.3334 MATLAB: R= 2.3449 %Difference= -0.49%	ANSYS: R= 2.2110 MATLAB: R= 2.2251 %Difference= -0.64%	ANSYS: R= 2.1589 MATLAB: R= 2.1880 %Difference= -1.35%
E4	300% Sink to Source Ratio	a=0.01 b=0.03	ANSYS: R= 1.6358 MATLAB: R= 1.6467 %Difference= -0.67%	ANSYS: R= 1.5628 MATLAB: R= 1.5875 %Difference= -1.58%	ANSYS: R= 1.5493 MATLAB: R= 1.6210 %Difference= -4.63%

From all of these simulations, several important points can be concluded about the analytical model developed by Ellison. First, the model needs to take the thickness and efficiency of the fins into account when calculating the effective heat transfer coefficient, h_e. This drastically increases the accuracy of the model under many scenarios. Second, the base thickness of the finned heat sinks has minimal effect on the accuracy of the model. Third, there is a heat spreading resistance along the fins. This spreading resistance along the fins can be an issue, but only in cases where the heat sinks are much longer than the area affected by the source. Reasonable scenarios see little effect from the spreading resistance because the fins extend minimally beyond the area affected by the heat source. Finally, in a very few cases, there is a slight spreading between fins but its affect is negligible. Fin efficiency and heat spreading along the fins seem to be the dominate factors affecting thermal resistance.

4.4 Heat Spreading Experimentation

4.4.1 Thermoelectric Power Unit Test Stand

The test stand used for experimental validation of the heat spreading phenomena consisted of a blower, inline heater, and counter flow heat exchanger, which is depicted in Figure 4.7. The blower begins the flow path of the air which continues through a knife valve. This regulates the flow rate of the air which is then measured by a flow meter. After the air passes through the flow

meter, it proceeds to the cold side of the heat exchanger inlet $T_{c,in}$ and passes through the top side of the heat exchanger which preheats the air. The air exits the heat exchanger cold side $T_{c,out}$ and continues through the inline heater where the temperature can safely be controlled and monitored up to 250°C. It then enters the hot side of the heat exchanger $T_{h,in}$ and exits it $T_{h,out}$. The air is then exhausted to atmosphere.

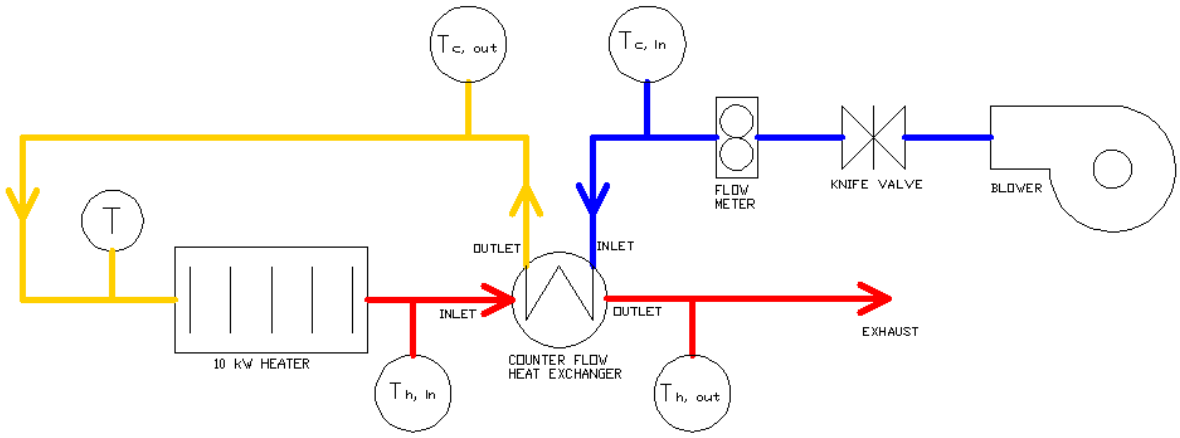


Figure 4.7: Schematic of experimental heated exhaust simulator test stand setup.

The original version of the test stand was developed to use water as the cooling fluid with air as the heat exchanger hot side fluid. This allowed for two separate loops of fluid as opposed to the heat exchanger being used in a preheat scenario as is done in this experiment. Smith's thesis elaborates more on the older versions of the test stand which can be seen in Figure 4.8 for comparison [9]. All of the experimentation in this thesis used the air-to-air counter flow heat exchanger shown in Figure 4.7 for validation of analytical and numerical results. Similarities between the older versions of the test stand and current version of the stand include the properties resulting from the abilities of the blower and inline heater. The blower operates up to 0.056 kg/s which equates to 100 standard CFM. The inline heater was ramped up and down at 5.4 °C/min to ensure the heater would not fail. Each component uses three-phase 208 Volt electricity as their power source for operation.

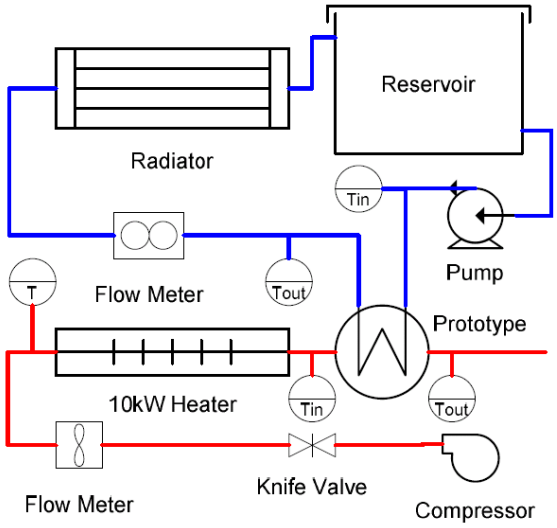


Figure 4.8: Schematic of an older version of experimental heated exhaust simulator test stand setup [9].

The primary differences between the older version of the test stand and the current version of the test stand were all related to flow configuration and the thermoelectric heat exchanger. The flow configuration was changed to create a preheat scenario with the air which acted as the cold side of the heat exchanger. Three valves were added to the flow path to allow the air to be manipulated into the direction of flow as needed. These valves allow for return to the older version of the test stand if water was ever to be used as the heat sink again. The other primary difference between versions of the test stand was the heat exchanger. The current version of the heat exchanger has a cold side and a hot side that are geometrically congruent to one another. There are four inlet/outlet manifolds that attach and detach from the main heat exchanger body. These connect the heat exchanger to the piping of the test rig. Each side of the heat exchanger has fins to increase heat transfer. These fins can be removed and another set of more or less aggressive fins can be used to change the heat transfer properties of the system. For the purposes of all testing done in this thesis, rectangular finned heat exchangers from M & M Metals [38] by the part number of MM10551 were used. These were cut and placed side by side in each side of the heat exchanger leaving 15 fins across the width of the heat exchanger. Each fin has a thickness of 1.91 mm (0.075 inches) and is spaced approximately 5.67 mm (0.235 inches) from adjacent fins. The length of the fin is 1 inch. The base of the store bought fin is 6.35 mm (0.250 inches) thick and in conjunction with the thickness of the shell of the heat exchanger, a base thickness of 12.32 mm (0.485 inches) is used for calculations.

Overall dimensions of the heat exchanger are 0.46 meters (18 inches) long by 0.127 meters (5 inches) wide. On each side of the heat exchanger, 12 thermocouples are placed into holes evenly spaced in the configuration shown in Figure 4.9. The figure specifically represents the hot side of the heat exchanger while the cold side thermocouples are on the bottom face of the cold side of the heat exchanger. These thermocouples are used to determine a zone temperature reading. A

zone is defined as an area in which the temperature is considered to be the same to simplify calculations pertaining to the behavior of thermoelectric devices. This is described more in depth later in Chapter 5 and Chapter 6. Thermocouples are used to record the inlet of the cold side of the heat exchanger ($T_{c,in}$), exit of the cold side of the heat exchanger ($T_{c,out}$), inlet to the inline heater (T), inlet to the heat exchanger hot side ($T_{h,in}$), and exit of the heat exchanger hot side ($T_{h,out}$). In addition to all of the thermocouple measurements, a mass flow reading is made by a flow meter approximately 10 pipe diameters downstream of the knife valve. All of this data is gathered by LabVIEW® and recorded into a file for processing. The data acquisition system is set up to reflect the configuration of the system which varies by test. Each test set up and additional recorded data will be explained in the following section.

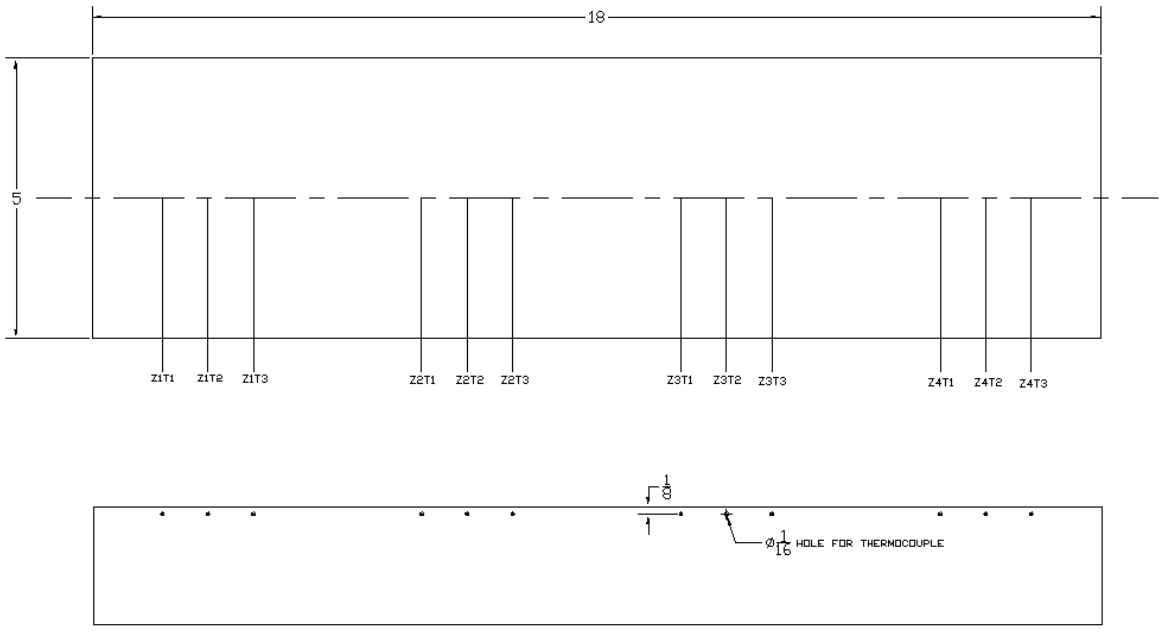


Figure 4.9: Thermocouple placement for the hot and cold side of the heat exchanger (thermocouple spacing not drawn to scale).

4.4.2 Heat Exchanger Characterization

4.4.2.1 Characterization Analysis

The initial phase of testing was performed to characterize the heat exchanger that had been designed for the test rig. This involved removing all thermoelectric modules placing the hot side of the heat exchanger in direct thermal contact with the cold side of the heat exchanger. Using basic principles of heat transfer from Incropera et al. [37], thermal resistance across the heat transfer surface can be determined and this baseline value can be used in later tests involving thermoelectric devices.

$$q_h = \dot{m}_h c_{p,h}\left(T_{h,i} - T_{h,o}\right) \quad (4.4.1)$$

$$q_c = \dot{m}_c c_{p,c}\left(T_{c,i} - T_{c,o}\right) \quad (4.4.2)$$

q represents the total heat transfer rate, $\dot{m}$ is the mass flow rate, c_p is the specific heat capacity at constant pressure; subscripts h and c represent the hot and cold sides of the heat exchanger respectively, and subscripts i and o specify inlets and outlets of the mass streams of temperature, T, respectively. These equations are independent of flow arrangement and heat exchanger type. In an ideal heat exchanger, these equations should equal each other as all of the heat is transferred from the hot side to the cold side of the heat exchanger. For real situations, some heat will be lost to the environment. A test was performed to characterize this phenomenon and the results are tabulated later in this section. These losses transformed equation (4.4.1) into equation (4.4.3).

$$q_h = \dot{m}_h c_{p,h}\left(T_{h,i} - T_{h,o}\right) - q_{losses} \quad (4.4.3)$$

As a result of equation (4.4.3), q_h is again equal to q_c which will now be noted as q. In addition to equations (4.4.2) and (4.4.3), another equation can be used to represent the simple heat exchanger model and eventually develop a total resistance for the heat exchanger which can be used throughout the remaining evaluations of each system setup.

$$q = UA_{HX}\Delta T_{lm} \rightarrow UA_{HX} = \frac{q}{\Delta T_{lm}} \quad (4.4.4)$$

where q is the heat transfer rate, UA_{HX} represents the overall heat transfer coefficient, and ΔT_{lm} is the log mean temperature differences which for a counter flow configuration can be calculated by

$$\Delta T_{lm} = \frac{\Delta T_2 - \Delta T_1}{ln(\Delta T_2/\Delta T_1)} = \frac{T_{h,o} - T_{h,i} + T_{c,o} - T_{c,i}}{ln\left(\frac{T_{h,o} - T_{c,i}}{T_{h,i} - T_{c,o}}\right)} \quad (4.4.5).$$

Solving for q using either equation (4.4.2) or (4.4.3), and ΔT_{lm} using the known temperatures of the heat exchanger system input into equation (4.4.5); equation (4.4.4) can be used to find a value for UA_{HX}. The reciprocal of this value will yield the total thermal resistance of the heat exchanger which will be used throughout later analysis.

$$R_{HX} = \frac{1}{UA_{HX}} \quad (4.4.6)$$

4.4.2.2 Characterization Experimental Results

Several tests were run with the goal of determining an accurate value for q_{losses} and R_{HX}. The initial tests raised many questions and concerns as to the ability of the system test rig, the accuracy of the measurements and testing methods, and the correctness of the analytical

methods. As testing commenced, the first tests used less insulation surrounding the heat exchanger and its manifolds. When the test results were analyzed, it was apparent that there was an extremely large energy loss from the system which would prohibit the calculation of the parameters of interest. More insulation was added to compensate for what was believed to be heat losses. Simultaneously, data was being acquired to help determine when steady state had been reached or the point at which properties were not changing or changing at a miniscule level to be considered approximately the same as the previous time step. The method determined if steady state had been reached by taking an average of the steady state variables in equations (4.4.1) and (4.4.2) over five time steps, except for the constant specific heat, and then dividing this result by the result from ten time steps prior. Each data point is taken 30 seconds apart from the previous point.

$$Steady\ State\ Eval = \frac{q_t}{q_{t-10}} \tag{4.4.7}$$

where

$$q_t = c_p \left(\frac{\sum_{t-4}^{t} \dot{m}}{5}\right)\left[\left(\frac{\sum_{t-4}^{t} T_i}{5}\right) - \left(\frac{\sum_{t-4}^{t} T_o}{5}\right)\right] \ or \ q_t = c_p \left(\frac{\sum_{t-4}^{t} \dot{m}}{5}\right)\left[\left(\frac{\sum_{t-4}^{t} T_o}{5}\right) - \left(\frac{\sum_{t-4}^{t} T_i}{5}\right)\right] \tag{4.4.8}$$

depending on which equation is being evaluated.

When steady state evaluation, equation (4.4.7), equals approximately 1.00 consistently, the system is said to have reached a steady state. Depending on the arrangement of the heat exchanger and additional components, i.e. aluminum blocks or thermoelectric modules, it would often take approximately two hours to reach steady state.

Another factor believed to be causing error in the results, was the accuracy of the thermocouples. Each of the thermocouples was calibrated previously. Thermocouple readings were taken in the four locations at the inlets and outlets of the heat exchanger, ($T_{c,in}$), ($T_{c,out}$), ($T_{h,in}$), and ($T_{h,out}$). Test results showed that these additional thermocouples were reading within a range that could be reasonable expected from the test setup. As a measure of insurance that the four primary thermocouples were reading accurately, they were each calibrated to the hot inlet location because it is the first reading taken downstream of the inline heater and the highest temperature reading that is made. Measurements were made at this location by each thermocouple at 40°C, 100°C, 150°C, 200°C, and 250°C after the system had reached steady state at each of these temperatures. The following tests performed found that there were still discrepancies occurring between expected results and actual results.

Because of the remaining discrepancy, mixing of the heat given off by the inline heater to the flow in an even distribution to the cross section of the pipe was called into question. A test was performed to check if there was in fact a temperature gradient across the inside of the pipe on the entrance to the hot side of the heat exchanger cross section. The inline heater was set to 200°C and steady state was reached. The thermocouple in the $T_{h,in}$ position was moved a quarter of an

inch from pipe wall to pipe wall and temperature readings were taken at each increment. The pipe diameter is two inches. Minor differences less than 2°C were observed between all of the increment locations with the highest temperature being in the approximate center of the pipe and the lowest temperatures being near the edges of the pipe. Because of this result, each thermocouple was adjusted to take readings as close to the center of the pipe as reasonably possible. It was also determined that this phenomenon had little to no impact on the final results because the slight differences were less than 1% of the values being measured. There was not enough inaccuracy in these readings to cause the severe discrepancies that were occurring throughout testing.

Remaining likely perceived causes for the inaccuracies were limited. Mass flow losses through each of the elbow joints, valves, flexible duct, and various other connections were of growing concern. The primary concern for the air escaping was through the flexible tubing that connects the pipes to the manifolds because many small leaks existed and the method of connecting them to the heat exchanger was not the most efficient. It was obvious that there were minor leaks in the system but these leaks had not been quantified due to lack of instrumentation. A high temperature silicone caulk was purchased in an effort to eliminate as many leaks as possible. The caulk appeared to be effective but the inaccuracies in the data still existed. Through the process of fixing the air leaks, it was noted that the flexible tubing seemed to still have issues with maintaining a leak free connection. To determine if air leaking was still an issue, a flow meter was acquired to perform some air tests. It was possible to adapt the flow path to meet the old method of operation where water jackets were used to cool the heat exchanger. This eliminates several stretches of pipe and the whole cold side of the heat exchanger. The values observed by the flow meter installed in the system downstream of the knife valve read approximately the same under each piping configuration. However, the additional flow meter was placed at the system exhaust to atmosphere and a considerable difference in flow rate was observed between the two piping configurations. Because the two flow meters were not calibrated to each other, it is hard to quantify an exact amount of air lost but it is safe to say that the primary reason discrepancies in the data existed was because of air leaks.

The main reason for the air leaks was because of the size and type of flexible duct. The size at the connection points to the heat exchanger was slightly too large providing small escape paths for the air. Also, the flexible dryer duct could not maintain its integrity as small holes were easily formed. New duct was ordered to replace the older metallic dryer duct. The new duct was a medium weight thermoplastic rubber hose with a spring steel wire helix rated for high temperatures which prevented small abrasions and kept air leaks to a minimum. In addition, this duct was sized properly for the connections made to the heat exchanger which drastically improved performance. Unfortunately, all of the air leaks could not be removed from the test rig. A method for compensating for the losses was added to the data analysis. The flow meter at the exhaust was used to measure the ratio of flow lost depending on system configuration. Using the counter flow setup, a reading would be taken at the exhaust. Using the water jacket setup,

another reading was taken at the exhaust. The first reading was then divided by the second, resulting in a ratio that was multiplied by the mass flow rate for the hot side because all of the air had passed through the flexible ducting at this point. The cold side mass flow was assumed to be a fairly accurate reading because most of the losses occurred when the configuration of the piping was set to create the counter flow heat exchanger. The ratio was almost always 0.94 and was changed in the analysis to reflect any differences from 0.94 in any tests that received different results than that. Now that the major system concerns for discrepancies in results were removed or compensated for, q_{losses} and R_{HX} could be found.

To find q_{losses}, the hot side of the heat exchanger was installed into the test rig. It was fully wrapped to provide approximately one inch of insulation. The test was run at 2.83 m^3/min (100 CFM) and $T_{h,in}$ equal to 200°C as these would be the testing conditions used for each of the heat spreading experiments. Analyzing the data points taken at steady state with equation (4.4.1), q_h can be considered to be q_{losses} and resulted in approximately 210 W. This value was the average of the calculations of all the steady state data. It can be assumed that some of the heat lost out of the hot side of the heat exchanger would escape into the cold side so a value of 190 W was used for the remaining analysis.

To find R_{HX}, both the hot and cold sides of the heat exchanger were placed into contact with each other. A layer of thermal paste was spread between the heat transfer surfaces to reduce contact resistance between the two heat exchanger faces. The whole heat exchanger unit was wrapped in insulation with the insulation covering all exposed surfaces. The heat exchanger system was allowed to reach steady state and analysis of the data taken during this time yielded a result for R_{HX}. To get R_{HX}, collected data with q_{losses} determined by the previous test was input into equations (4.4.2), (4.4.3), and (4.4.5). These equations enabled the calculation of q_c, q_h, and ΔT_{lm}, respectively. Theoretically, q_h and q_c should be equal but their results varied from each other slightly throughout all tests. This is partially the result of inaccuracy of data acquisition devices and the inaccuracy of q_{losses} being propagated through each experiment. Because the application of insulation needed to be repeated for each experiment, it may have slightly skewed the value of q_{losses} for each test resulting in some error. Plugging both q_h and q_c into equation (4.4.4), two results were obtained for UA_{HX}. Each of these results was used in equation (4.4.6), to get two values for R_{HX} which were averaged together. This yielded the result for R_{HX} equal to 0.0644 K/W. R_{HX} = 0.0644 K/W is used throughout the remaining analysis for each experimental setup.

4.4.3 Heat Spreading Experimental Results

Several tests were run to capture the effect of heat spreading within the system. Each of the tests required a piece or multiple pieces of aluminum bar stock of varying dimensions to be placed in between the hot and cold sides of the heat exchanger. As a reminder, the dimensions of the area of the face of the heat exchanger was 0.46 meters (18 inches) long by 0.127 meters (5 inches) wide. The aluminum bar stock was chosen for its known thermal resistance as well as its ability

to mimic the analytical methods and solutions provided in section 4.2. Insulation was placed around the bar stock between each side of the heat exchanger to represent the boundaries provided by Ellison's model [14], as well as to direct the heat transfer through the bar stock. The six experiments that were run each used 25.4 mm (1 inch) thick bar stock but the length and width of the bar stock was varied as well as the number of pieces. The pieces of bar stock used for each test can be seen in Table 4.6. This table displays the number of bars used in the experiment, the dimensions of each piece of bar stock used, the cross sectional area of a single bar, and the total area of pieces of bar stock used. The orientation of the heat exchanger and bar stock can be seen in Figure 4.10 through Figure 4.15. Each test was performed at 200°C and 2.83 m^3/min (100 CFM) to remain consistent with the previous results for R_{HX}.

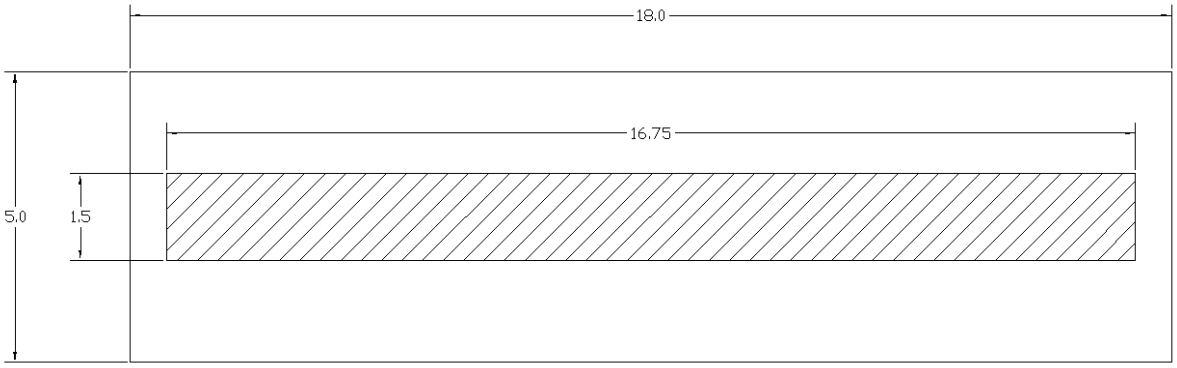


Figure 4.10: Test setup A, one aluminum bar, 1.5 inches x 16.75 inches.

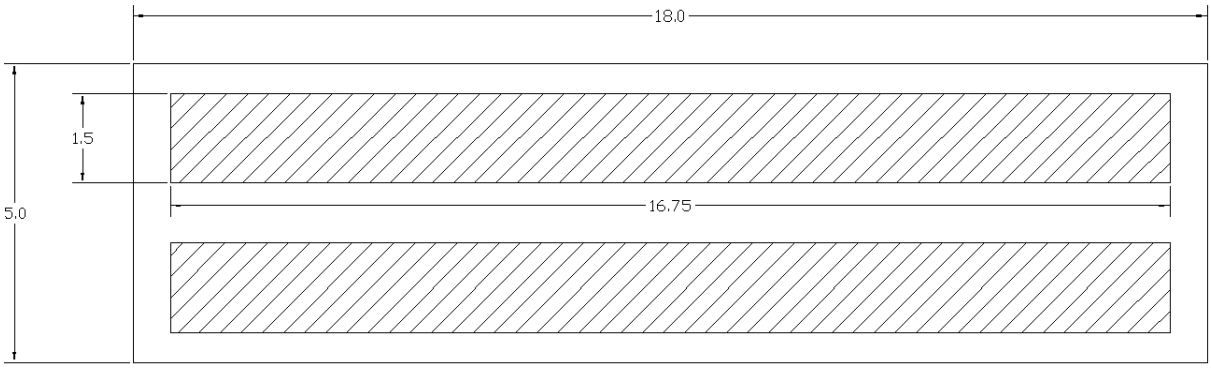


Figure 4.11: Test setup B, two aluminum bars, each 1.5 inches x 16.75 inches.

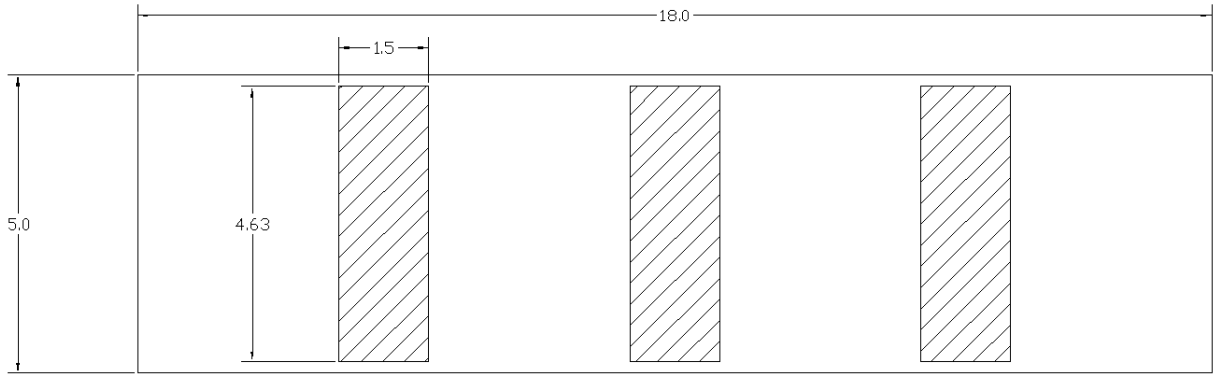


Figure 4.12: Test setup C, three aluminum bars, each 4.63 inches x 1.5 inches.

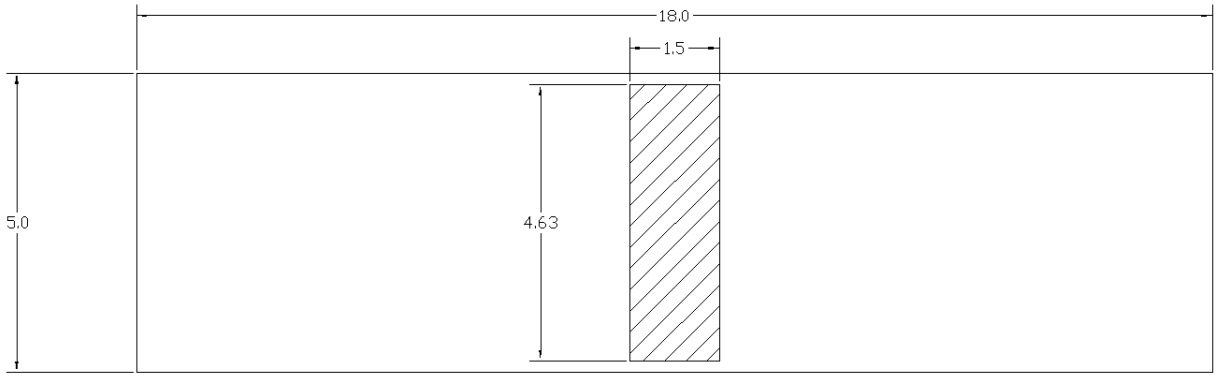


Figure 4.13: Test setup D, one aluminum bar, 4.63 inches x 1.5 inches.

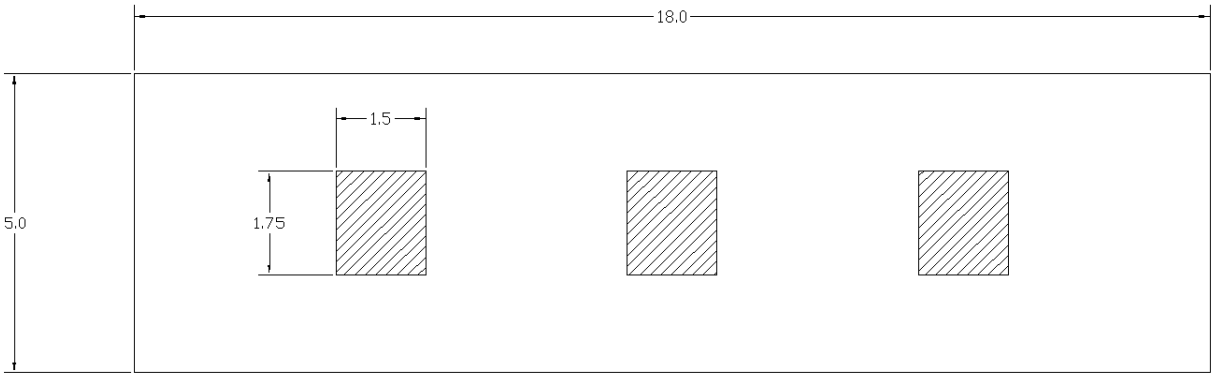


Figure 4.14: Test setup E, three aluminum bars, each 1.75 inches x 1.5 inches.

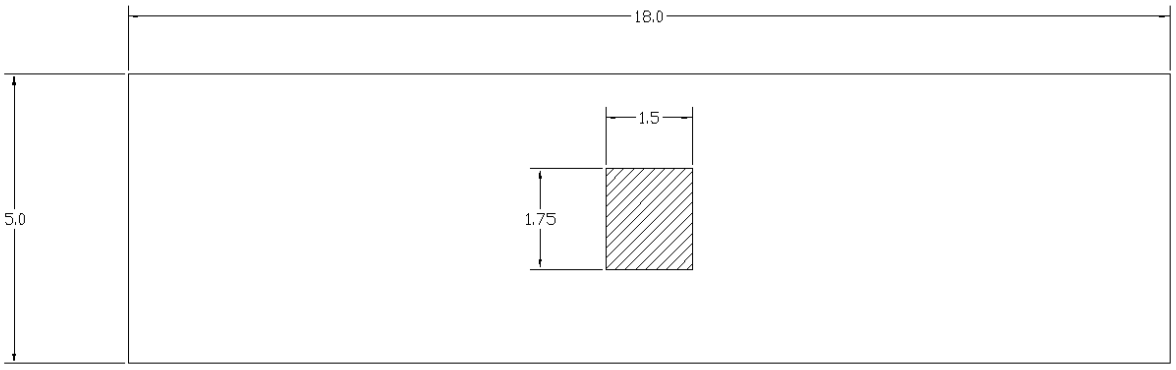


Figure 4.15: Test setup F, one aluminum bar, 1.75 inches x 1.5 inches.

Table 4.6: Heat Spreading Test Configurations.

Test	*Number of Bars*	*Dimensions*	*Area of a Single Bar [m^2]*	*Total Area of Bars [m^2]*
A	1	1.5 in x 16.75 in	0.01621	0.01621
B	2	1.5 in x 16.75 in	0.01621	0.03242
C	3	4.625 in x 1.5 in	0.00448	0.01343
D	1	4.625 in x 1.5 in	0.00448	0.00448
E	3	1.75 in x 1.5 in	0.00169	0.00508
F	1	1.75 in x 1.5 in	0.00169	0.00169

A similar method was used to find important variables as was used for R_{HX}. Because of the uncertainty in the result of q_{losses}, equation (4.4.3) is used strictly as a comparison for equation (4.4.2) to ensure that the results fall within the expected range. q_c is found using equation (4.4.2), which is then used in equation (4.4.9) to find $(UA)_{Tot}$.

$$q = (UA)_{Tot}\Delta T_{lm} \rightarrow (UA)_{Tot} = \frac{q}{\Delta T_{lm}} \tag{4.4.9}$$

This value is then used in equation (4.4.10) where the reciprocal is taken of the average heat transfer coefficient to find the total thermal resistance of the heat exchanger. R_{Tot} includes all resistances internal to the heat exchanger which include convection and conduction of the heat exchanger itself, the thermal resistance of the bar stock, heat spreading resistance, and resistance of the thermal paste. The resistance of the thermal paste will be addressed later in this section as it drastically skewed initial results and complicated analysis.

$$R_{Tot} = \frac{1}{(UA)_{Tot}} \tag{4.4.10}$$

A direct comparison can be made between R_{Tot} and the total area of the bar stock for each test. Table 4.7 is shown for comparison purposes.

Table 4.7: Comparison of R_{Tot} and Total Area of the Aluminum Bar Stock.

Test	R_{Tot} *[K/W]*	*Total Area of Bars* $[m^2]$
A	0.0916	0.01621
B	0.0752	0.032419
C	0.1069	0.013427
D	0.2717	0.004476
E	0.2547	0.005081
F	0.3556	0.001694

In this table, it can be seen that the test with the least area of bar stock has the highest total thermal resistance and vice versa. This is because the insulation surrounding the bar stock provides additional resistance to the heat exchanger as a whole which is reflected in the values for the total thermal resistance.

Initial testing and analysis did not account for the impact of the resistance of the thermal paste used to eliminate or minimize contact resistance. When the thermal paste is not accounted for, the general method to obtain the spreading resistance of the aluminum bar faces to the heat exchanger faces is equal to R_{Tot} minus R_{HX} minus the resistance of the aluminum bar. More specifically, equation (4.4.11) represents the actual method for finding the spreading resistance for one face of an aluminum bar to one side of the heat exchanger.

$$R_{Sp} = \frac{1}{2} N_{bars}(R_{Tot} - R_{HX} - R_{bar}) \tag{4.4.11}$$

R_{Sp} represents the spreading resistance, N_{bars} is the number of aluminum blocks sandwiched between the heat exchanger, R_{Tot} is the total thermal resistance for the particular setup of the experiment, R_{HX} is the heat exchanger thermal resistance as determined in a previous test, and R_{bar} is the thermal resistance of all of the aluminum bar stock in each experiment. The reason that the ½ exists in the equation is because the spreading resistance occurs on both sides of the heat exchanger. However, R_{Sp} only represents the spreading for a single face to face location of the bar stock to the heat exchanger. N_{bars} is used to represent the number of pieces of bar stock between sides of the heat exchanger because each instance of spreading is considered to be exclusive from the other instances. This is done to simplify analysis and avoid double counting of important parameters. R_{bar} is the conductive resistance of the aluminum bar stock and uses the standard resistance equation, (4.4.12), manipulated to fit the model. This value is calculated in each test depending on the size of the bar stock in that experiment and the number of bars used.

$$R_{bar} = \frac{t_{bar}}{kA_{bar}N_{bars}} \tag{4.4.12}$$

The variable A_{bar}, the cross sectional area of the piece of bar stock, is the only one that changes depending on the experiment whereas t_{bar}, the thickness of the bar stock, is always 1 inch or 0.0254 m, and k, the thermal conductivity, is approximately 180 W/(m-K) for this type of aluminum. N_{bars} is used to represent the bars in thermal parallel in their configuration. When equation (4.4.12) is inserted into equation (4.4.11), variables cancel and the resistance of just one bar is used to find the spreading resistance.

All of the aforementioned analysis yielded the results seen in Table 4.8.

Table 4.8: Initial Heat Spreading Experimental Results Compared to Analytical Solutions.

Test	*Number of Bars*	R_{Tot} *[K/W]*	R_{bar} *[K/W]*	R_{Sp} *Exp [K/W]*	R_{Sp} *Analytical [K/W]*	*Exp/Analy*
A	1	0.0916	0.00871	0.00925	0.0052	1.78
B	2	0.0752	0.00435	0.00647	0.0016	4.08
C	3	0.1069	0.01051	0.04796	0.0231	2.07
D	1	0.2717	0.03153	0.08789	0.0610	1.44
E	3	0.2547	0.02777	0.24383	0.0539	4.53
F	1	0.3556	0.08332	0.10400	0.0922	1.13

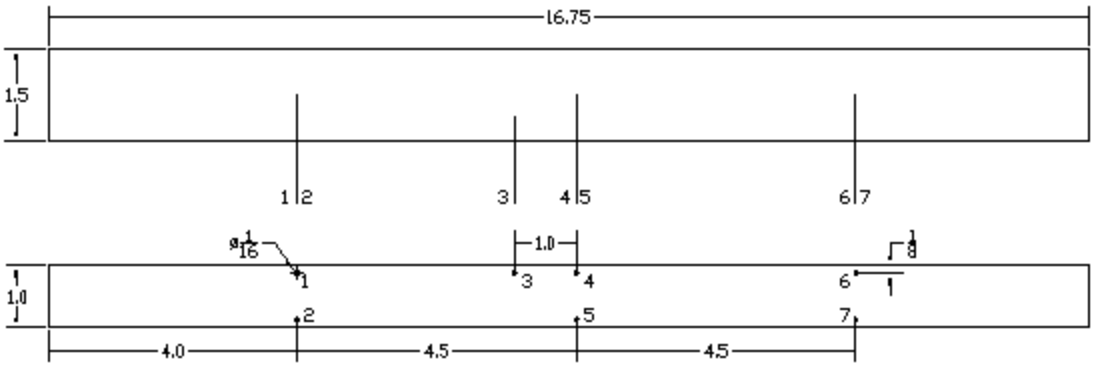
The analytical result was found using the method described in section 4.2. The convective heat transfer coefficient, h, was set to 85.887 W/(m^2-K). This was calculated using the experimental value for R_{HX}, and balancing this with the calculated conductive resistances for the shell of the heat exchanger and the paste, and equations (4.2.19) and (4.3.1-4.3.6) which represented the convection throughout the heat exchanger and the conduction through the fins. A nonlinear solver found the solution for the heat transfer coefficient, h, at a value of 85.887 W/(m^2-K).

It should be noted that almost every experiment produced a spreading resistance considerably larger than the analytical spreading resistance. Many were 2-4.5 times larger meaning that another factor was contributing to the inflated experimental spreading resistance. This factor stems from the concept of thermal contact resistance. Thermal contact resistance occurs because of surface roughness between faces. In these experiments, *OMEGATHERM 201* paste from Omega Engineering, Inc. [39] was used to reduce the thermal contact resistance to a negligible effect. Unfortunately, the paste still provided a substantial amount of conductive thermal resistance which was skewing the effect of the heat spreading. Steps were taken to quantify a value for the contact resistance, R_{con}, with the thermal paste in effect.

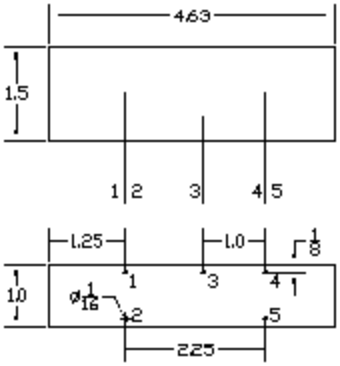
The initial attempt to quantify the thermal resistance of the thermal paste involved an attempt to find the conductive resistance of a layer of paste. Manufacturer datasheets list the conductivity, k_{paste}, as 16 (BTU-in)/(hr-ft^2-°F). Converting units shows that this value is equal to 2.308 W/(m-K). However, the thermal resistance for conduction was difficult to determine because the thickness of the layer of paste was immeasurable. Although it was a very thin layer in each of the

tests, this thin layer varied in thickness from test to test. A wide range of reasonable estimates was used but it was impossible to determine which result was the most accurate. Also, as the heat exchanger is assembled together, the starting thickness of the paste changes because of compression. In short, there is no definitive way of knowing the thickness of the paste layer.

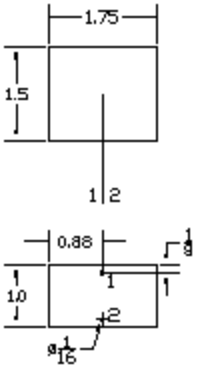
There were many thermocouples used in data collection. As it turned out, several thermocouples were helpful in determining reasonable estimates for the contact resistance which includes the conduction resistance caused by the paste. For each test with the aluminum bars, thermocouples were inserted into holes in the side of the aluminum block created for the thermocouples. These holes marked approximately the center of the face of each block. Depending on the size of the block, there were between two and seven thermocouples. Each thermocouple was 3.18 mm (1/8 inch) from the face that they were measuring the temperature of. See Figure 4.16 A-C for the configuration used in each test. Each of the thermocouples was inserted approximately to the center of the aluminum blocks with the exception of each number “3”. Number “3” for each block was set closer to the edges of the blocks to determine if there were any edge effects. No significant edge effects were detected.



(A)



(B)



(C)

Figure 4.16 A-C: (A) Used in Test A and Test B. (B) Used in Test C and D. (C) Used in Test E and F.

Measurements were taken 3.18 mm (1/8 inch) from the adjacent faces of the heat exchanger. The conductive resistance per unit area through the two 3.18 mm (1/8 inch) portions of aluminum was subtracted from the change in temperature between thermocouples divided by the heat rate as determined by equations (4.4.2) and (4.4.3) for the respective sides divided by the area to obtain heat flux. This left the thermal resistance per unit area of the thermal grease. These calculations were made for every data point of every available thermocouple in the aluminum bars. Average values were taken and a hot side contact resistance per unit area, $R''_{con,h}$, and a cold

side contact resistance per unit area, $R''_{con,c}$, were determined. The final result for these were implemented into equation (4.4.11) resulting in equation (4.4.13).

$$R_{Sp} = \frac{1}{2} N_{bars} \left(R_{Tot} - R_{HX} - R_{bar} - \frac{R''_{con,c}}{A_{bar,tot}} - \frac{R''_{con,h}}{A_{bar,tot}} \right) \quad (4.4.13)$$

$A_{bar,tot}$ represents the total area of each of the aluminum bars on one face of the heat exchanger. Results using this equation can be seen in Table 4.9.

Table 4.9: Heat Spreading Experimental Results Using the Measured Contact Resistance Per Unit Area Results Compared to Analytical Solutions.

Test	*Number of Bars*	R_{Tot} *[K/W]*	R_{bar} *[K/W]*	R_{Sp} *Exp w/measured* R''_{con} *[K/W]*	R_{Sp} *Analytical [K/W]*	*Exp/Analy*
A	1	0.0916	0.00871	0.00589	0.0052	1.13
B	2	0.0752	0.00435	-0.00707	0.0016	-4.46
C	3	0.1069	0.01051	0.02337	0.0231	1.01
D	1	0.2717	0.03153	0.02984	0.0610	0.49
E	3	0.2547	0.02777	0.15788	0.0539	2.93
F	1	0.3556	0.08332	0.04699	0.0922	0.51

Using the measured contact resistances per unit area for the cold and hot sides in each experiment provided some improvement in the results for the spreading resistance in most cases except Test B. In the case of Test B, the spreading resistance yielded a negative number when using the analysis of equation (4.4.13). This is because the spreading resistance was overestimated during this experiment. It is likely because the method used to find $R''_{con,h}$ and $R''_{con,c}$ is a good estimate in most cases but the thickness of the layer of paste can vary across the contact surfaces. Also, the thermocouples on either side of the paste do not line up directly so the change in temperature between the thermocouples may not directly reflect the correct temperature difference. A final reason for some error is a result of the heat spreading effect. It's difficult to assess the effect that the heat spreading has on the temperature readings which means that using these temperatures provides only a mediocre estimate at best.

To try to remove some of the variability in each test's thickness of its layer of paste, an average contact resistance per unit area needed to be calculated. The contact resistances per unit area for each test varied significantly and on an order of magnitude so the average of the contact resistances per unit area, $R''_{con,avg}$, from Test A was used. Per test A, the average value of 5.45E-5 (m^2-K)/W was calculated. This value was then used in equation (4.4.14) which is similar to equation (4.4.13). The only difference is that the contact resistance per unit area is taken to be a single value for both sides of the heat exchanger.

$$R_{Sp} = \frac{1}{2} N_{bars}(R_{Tot} - R_{HX} - R_{bar}) - N_{bars} \frac{R''_{con,avg}}{A_{bar,tot}} \quad (4.4.14)$$

The use of this equation and the average contact resistance per unit area improved results significantly in some cases and slightly made them worse in others. Results are displayed in Table 4.10.

Table 4.10: Heat Spreading Experimental Results Using the Average Contact Resistance Per Unit Area Equal to 5.45E-5 (m^2-K)/W Results Compared to Analytical Solutions.

Test	*Number of Bars*	R_{Tot} *[K/W]*	R_{bar} *[K/W]*	R_{Sp} *Exp w/* $R''_{con,avg}$ *[K/W]*	R_{Sp} *Analytical [K/W]*	*Exp/Analy*
A	1	0.0916	0.00871	0.00589	0.0052	1.13
B	2	0.0752	0.00435	0.00311	0.0016	1.96
C	3	0.1069	0.01051	0.03579	0.0231	1.55
D	1	0.2717	0.03153	0.07571	0.0610	1.24
E	3	0.2547	0.02777	0.21165	0.0539	3.93
F	1	0.3556	0.08332	0.07183	0.0922	0.78

When compared to Table 4.9, the ratio of results for experimental to analytical of Test A is the same because the contact resistances per unit area from that test were averaged. Test B improved from the previous model where its contact resistance was being drastically overestimated. Test E remains with unfavorable results which is likely the result of a considerably thicker layer of paste or inaccurate temperature readings.

One final effort was made to change the results to make it more capable to observe trends. If $R''_{con,avg}$ were to equal 6.5E-5 (m^2-K)/W, then the experimental results from Test A would line up very well with the analytical solution for that scenario. This value is in the reasonable range of measured contact resistances per unit area and can be used in the other tests for comparison. Table 4.11 shows the results from this analysis.

Table 4.11: Heat Spreading Experimental Results Using the Average Contact Resistance Per Unit Area Equal to 6.5E-5 (m^2-K)/W Results Compared to Analytical Solutions.

Test	*Number of Bars*	R_{Tot} *[K/W]*	R_{bar} *[K/W]*	R_{Sp} *Exp w/* $R''_{con,avg}$ *[K/W]*	R_{Sp} *Analytical [K/W]*	*Exp/Analy*
A	1	0.0916	0.00871	0.00524	0.0052	1.01
B	2	0.0752	0.00435	0.00246	0.0016	1.55
C	3	0.1069	0.01051	0.03344	0.0231	1.45
D	1	0.2717	0.03153	0.07336	0.0610	1.20
E	3	0.2547	0.02777	0.20545	0.0539	3.81
F	1	0.3556	0.08332	0.06562	0.0922	0.71

When compared to Table 4.10, it can be seen that all tests appear to improve except for Test F. Test F experimental result to analytical solution ratio slightly decreases because the contact resistance per unit area is slightly greater. This causes a drop in the calculated spreading resistance from the data which affects the value from the analytical solution. In general, if one aluminum bar is used in the set up, then it is likely to be more accurately predicted by the analytical model as can be seen in this table. The results for Test E are still very inaccurate. Part of this can be explained by the contact resistance being improperly identified for the exact setup used. Too much variability occurs with the paste and with the smaller blocks being used; the thermal contact resistance can dominate the results.

Overall, results in Table 4.9, Table 4.10, and Table 4.11 compared to results in Table 4.8 show that thermal contact resistance exists due to the layers of thermal paste between the aluminum bars and the heat exchanger faces. It must be compensated for as it was in the results shown in Table 4.9, Table 4.10, and Table 4.11. The results in Table 4.9 would seem to have the most promise for yielding accurate results but does not. This is because the placement of thermocouples do not have a direct correspondence for position on the heat exchanger and the bar stock. The variability and estimation that exists in the prediction of the thermal contact resistance per unit area is too great to consider these results accurate for every test. In most of the cases, the thermal contact resistance per unit area is overcompensated for, which diminishes the relative magnitude of the spreading resistance. Using an average of the most accurate evaluation of the thermal contact resistance from Test A, as was done with the results in Table 4.10, seems to be the best approach. The results display that the spreading resistance does exist and that it can be somewhat accurately measured. However, the variability of the thermal contact resistance still hinders the precision of the results. Changing the thermal contact resistance to reflect its expected value based on the analytical results, as was done with the results in Table 4.11, provides an improved analysis of the results. It is shown that the better prediction of contact resistance will strengthen the observation of the spreading resistance. Acceptable results were obtained with regards to the existence of spreading resistance and the accuracy of the analytical method for finding the spreading resistance. However, a better method for quantifying contact resistance is needed to fully validate the accuracy of the model and provide additional knowledge for further improvements to the model.

5 Thermoelectric Power Unit System Model

Developing an accurate, robust thermoelectric power unit system model is of great importance to TEPSS. It is significant for use throughout the industry for prediction of power based on the many available inputs into the system. With the advancements in the thermoelectric generators, niche heat recovery applications are brought to the forefront. A system model that accounts for several options for inputting TEG parameters, heat exchanger type selection, fin type selection, size constraints and assembly selection, and three dimensional heat spreading factored in allows for a sound prediction of power in a proposed system. The basic principle behind the system model is the balance of heat energy throughout the system taking into account the thermoelectric phenomena of heat conversion to electrical energy.

5.1 System Model Development

5.1.1 Basic Heat Exchanger Analysis

To evaluate the performance of a heat exchanger, total heat transfer rate must be related to inlet and outlet temperatures of the fluids, flow rates of the fluids, heat transfer coefficient, and surface area between hot and cold sides of the heat exchanger. Assuming that the fluids are not undergoing a phase change and constant specific heat is maintained, equations (4.4.1) and (4.4.2) can be used. Figure 5.1 represents the proposed basic heat exchanger for this introduction to the analysis. In this case, q_h equals q_c which is set equal to the total heat rate, q. These equations are also equal to equation (4.4.4) which utilizes equation (4.4.5) for counter flow heat exchanger analysis. Counter flow type heat exchangers will be used for the remainder of the system model development unless otherwise noted. Figure 5.2 shows the temperature distributions for a general counter flow heat exchanger. Because of the simplicity of this system, it can be solved easily. Three equations with three unknowns, likely the two outlet temperatures and the total heat transfer rate, is a simple system. Adding the thermoelectric generation equations slightly complicates the analysis and requires the addition of several equations.

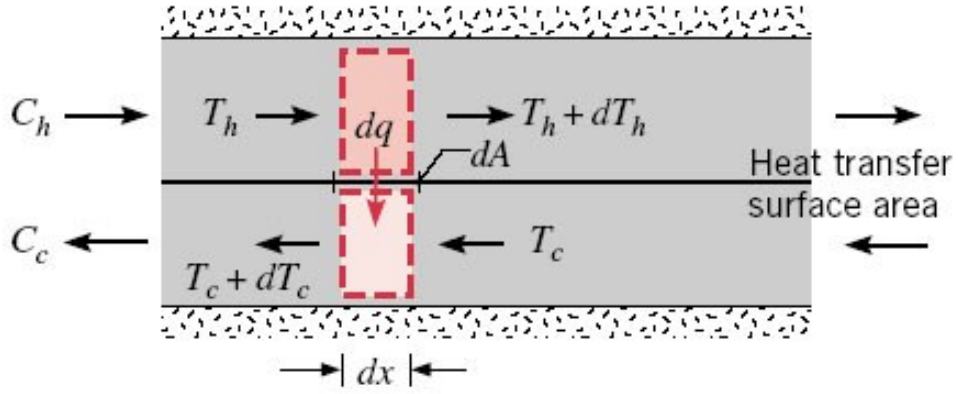


Figure 5.1: Basic counter flow heat exchanger [37].

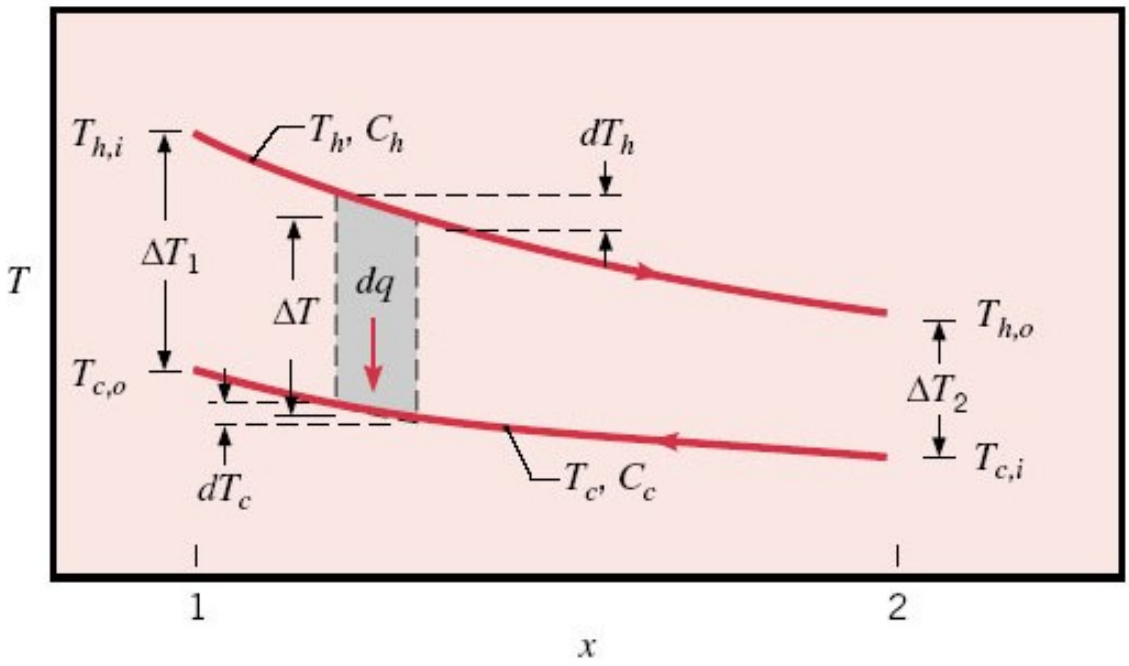


Figure 5.2: Temperature distributions for a basic counter flow heat exchanger [37].

5.1.2 Adding the Thermoelectric Generation Equations to the Heat Exchanger Analysis

A counter flow heat exchanger will work approximately the same as a heat exchanger that utilizes thermoelectric devices. The difference is that some of the heat energy is extracted from the system and converted to electrical energy by means of the TEG's. Also, the simple balance of heat energy in the previous section no longer exists as all the heat is not transferred from one side

of the heat exchanger to the other. In addition the heat rates will be dependent on the electrical current. The end goal of the derivation will result in a model that can be characterized by Figure 5.3. This figure represents the complex energy balance and progression of energy flow through a set number of zones dividing up the heat exchanger. A zone is a discrete area that encompasses the subsequent equations as well as inputs for all parameters. A zone is considered to be isothermal for the planes perpendicular to the heat flow through the thermoelectric modules. Each zone is solved for which eventually provides a final solution through an iterative process. As the zones are stepped through, some of the values for the next zone are provided from the previous zone such as flow temperatures. The performance of each zone is summed together to describe the overall performance of the heat exchanger. This will be described in section 6.2 where the TEPSS open source code is discussed more in depth.

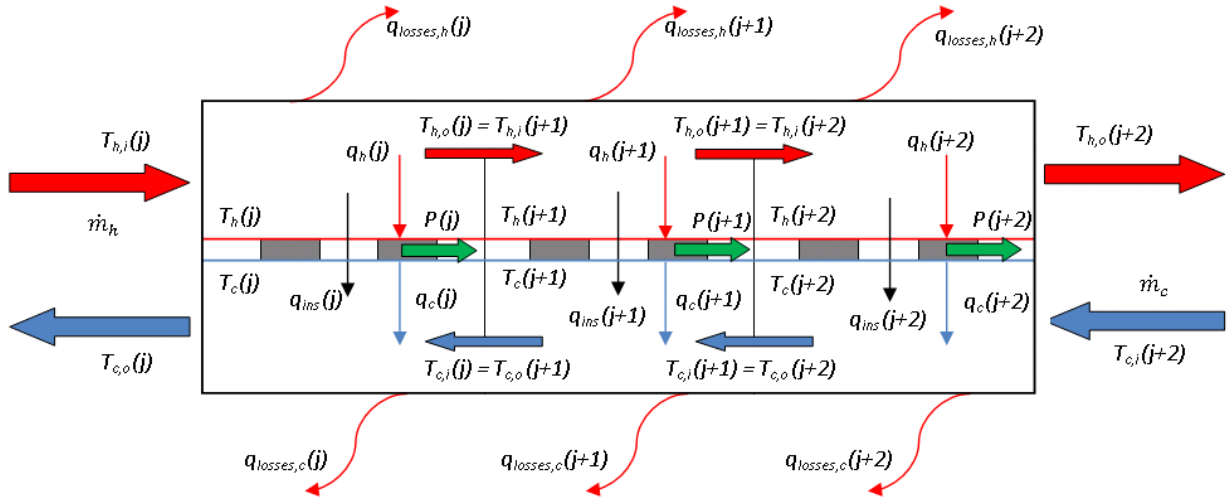


Figure 5.3: Thermoelectric power unit energy balance and TEPSS zone progression.

As a start to developing the thermoelectric power unit, thermoelectric equations need to be added to the simple heat exchanger from the previous section. The subsequent energy balance is for a single zone and can be expanded for an unlimited number of zones. The thermoelectric equations include (1.1.3), (1.1.4), (5.1.1), and (5.1.2). Equation (5.1.1) is shown as a reminder that power, *P*, is the difference between the hot and cold side heat rates.

$$P = q_h - q_c \tag{5.1.1}$$

$$I = \frac{\alpha(T_h - T_c)}{R_e + R_L} \tag{5.1.2}$$

Equation (5.1.2) demonstrates the relationship for electric current in a thermoelectric module. This is considered the load current based on the temperatures experienced by the module, the electrical related key parameters, and the load resistance across the module(s). The value for thermoelectric power is greatest when the internal electrical resistance equals the load resistance

Equation (5.1.2) is substituted into (1.1.3) and (1.1.4) resulting in equations (5.1.3) and (5.1.4) therefore removing the unknown current from the model.

$$q_h = \alpha\left[\frac{\alpha(T_h - T_c)}{R_e + R_L}\right]T_h + \frac{T_h - T_c}{R_{th}} - \frac{1}{2}R_e\left[\frac{\alpha(T_h - T_c)}{R_e + R_L}\right]^2 \quad (5.1.3)$$

$$q_c = \alpha\left[\frac{\alpha(T_h - T_c)}{R_e + R_L}\right]T_c + \frac{T_h - T_c}{R_{th}} + \frac{1}{2}R_e\left[\frac{\alpha(T_h - T_c)}{R_e + R_L}\right]^2 \quad (5.1.4)$$

Seebeck coefficient (α), module thermal resistance (R_{th}), and module electrical resistance (R_e) are considered known values as they are all properties of the thermoelectric and can be evaluated or determined by methods mentioned in section 5.2. Hot side surface temperature (T_h) and cold side surface temperature (T_c) are considered unknown in the system of equations that are developing as well as power (P), hot side heat rate (q_h), and cold side heat rate (q_c).With the modules in place, q_h and q_c are no longer equal to each other. There are now five unknowns with only three equations, (5.1.1), (5.1.3), and (5.1.4), meaning that this system is not solvable. Taking information known about the heat exchanger and balancing the energy increases the number of equations.

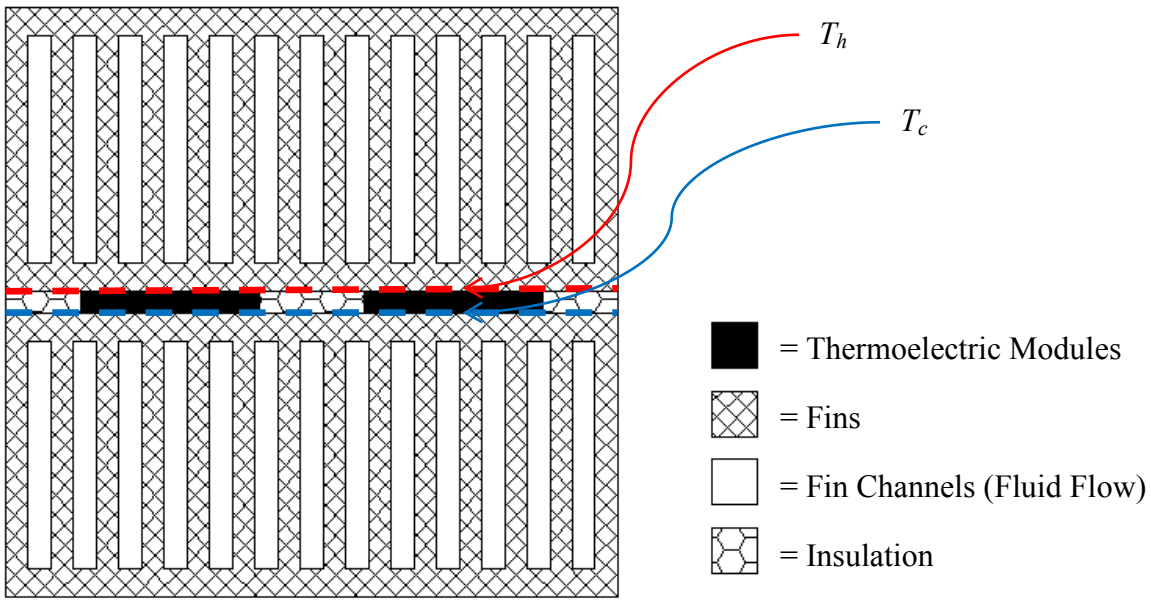


Figure 5.4: Thermoelectric power unit cross section.

As depicted in Figure 5.4, a layer of insulation exists around the thermoelectric module. This insulation has the same temperature difference across its surfaces, T_h minus T_c. The thermal bypass losses through the insulation can be modeled using a simple 1D resistance as

$$q_{ins} = \frac{T_h - T_c}{R_{ins}} \quad (5.1.5).$$

q_{ins} represents the heat transfer through the insulation and R_{ins} is the thermal resistance of the insulation. The insulation's thermal resistance can be calculated based on its conductive coefficient, its surface area, and the thickness of the modules. Therefore, R_{ins} is a known value given a set of design parameters. However, q_{ins} is unknown resulting in another variable for the system meaning that there are four equations with six unknown values.

Balancing energy around T_h, heat is transferred in through the flow energy and out through the insulation and thermoelectric layer with an intermediary resistance which is seen in Figure 5.5. The energy balance results in equation (5.1.6).

$$q_h + q_{ins} = \dot{m}_h C_{p,h}\left(T_{h,i} - T_{h,o}\right) \tag{5.1.6}$$

q_h and q_{ins} are the same as defined before while $\dot{m}_h$ is the mass flow rate of the hot side, $C_{p,h}$ is the specific heat capacity, which is assumed to be constant within each zone, of the hot side fluid, $T_{h,i}$ is the inlet temperature to the hot side flow, and $T_{h,o}$ is the outlet temperature to the hot side flow. This is for a single zone as the energy balance is described for a single zone only. $T_{h,o}$ becomes $T_{h,i}$ for the next zone downstream when the model is expanded to a multi-zone heat exchanger.

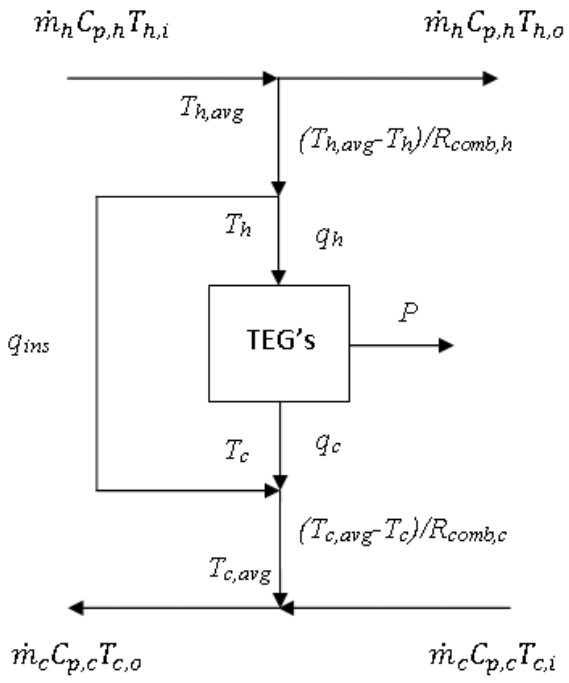


Figure 5.5: Energy balance circuit of initial model for a zone.

A similar tactic can be used for the cold side of the heat exchanger. Energy is removed from the system in the form of electrical power as seen in equation (5.1.1). The remaining heat energy is transferred to the cold side of the thermoelectric module T_c through the module and the

insulation. This energy that is transferred to the cold side is equal to the flow energy on the cold side as seen in equation (5.1.7).

$$q_c + q_{ins} = \dot{m}_c C_{p,c}\left(T_{c,o} - T_{c,i}\right) \tag{5.1.7}$$

q_c and q_{ins} are the same as previously defined, $\dot{m}_c$ is the cold side mass flow rate, $C_{p,c}$ is the specific heat capacity of the cold side fluid, and $T_{c,o}$ and $T_{c,i}$ are the outlet and inlet temperatures, respectively, to the cold side flow.

Both $\dot{m}_h$ and $\dot{m}_c$ are considered known properties as they can be controlled or measured by the model user. $C_{p,h}$ and $C_{p,c}$ are also considered known properties because considerable information is known for a multitude of fluids at varying temperatures. They are also calculated by TEPSS based on fluid properties and then input into the system of equations but this is discussed later in this chapter. This leaves the four temperatures in equations (5.1.6) and (5.1.7). They are treated as if one temperature from each equation is known and one from each equation is unknown. This is because the TEPSS solver can obtain a solution with either temperature as the equations will balance with known values and unknown values. For this analysis, $T_{h,i}$ and $T_{c,o}$ are considered known values, and $T_{c,i}$ and $T_{h,o}$ are the unknowns. This has introduced two more equations and two more unknowns for a total of six equations with eight unknowns.

The average flow temperature for both the hot side, $T_{h,avg}$, and cold side, $T_{c,avg}$, can provide a powerful value for solving the system. Equations for finding average flow temperatures are seen in (5.1.8) and (5.1.9).

$$T_{h,avg} = \frac{T_{h,i}+T_{h,o}}{2} \tag{5.1.8}$$

$$T_{c,avg} = \frac{T_{c,o}+T_{c,i}}{2} \tag{5.1.9}$$

There are now eight equations and ten unknowns making the system unsolvable still. Therefore, additional equations are needed to relate the variables and provide an independent solution for the unknown variables. The introduction of the average inlet and outlet temperatures allow for additional equations to be developed describing the system. The heat energy transferred from $T_{h,avg}$ to T_h is the same heat energy entering into the hot side of the thermoelectric modules and insulation. The thermal resistance between these two temperatures is represented by $R_{comb,h}$ and can be determined based on the heat transfer coefficient and possibly fin geometries that exist in this heat flow path. Equation (5.1.10) represents the portion of the system that was just described.

$$q_h + q_{ins} = \frac{T_{h,avg}-T_h}{R_{comb,h}} \tag{5.1.10}$$

Similar analysis is provided for the cold side of the heat exchanger. T_c represents the temperature of the cold side of the thermoelectric module and insulation while $T_{c,avg}$ represents the cold side

average flow temperature. The difference of these temperatures divided by the thermal resistance on the cold side of the heat exchanger is equal to the heat energy transferring through the cold side of the heat exchanger and can be seen in equation (5.1.11).

$$q_c + q_{ins} = \frac{T_c - T_{c,avg}}{R_{comb,c}} \tag{5.1.11}$$

With the equations provided, the unknown variables in the system become independent of each other. The system is now solvable but improvements can be made to this set of equations to be more accurate and robust. One such improvement is removing the average flow temperatures and replacing them with temperatures determined by a log mean temperatures difference.

5.1.3 Utilizing a Log Mean Temperature Difference

The thermoelectric power unit is currently represented by a system of ten equations which consists of equations (5.1.1) and (5.1.3)-(5.1.11).This set of equations works particularly well when a small amount of heat is removed from the working fluid. This is because the surface area is approximately isothermal which means that the average temperature found using equations (5.1.8) and (5.1.9) is sufficient. However, if the surface area grows larger, the analysis begins to break down because the surface is no longer isothermal. Modules may vary in size or the number of modules represented by this analysis could potentially increase. Also, the thermoelectric module could be surrounded by a considerable amount of insulation which will increase the surface area and distort the average value. The average temperature is a less accurate depiction of the temperatures as they relate between zones.

A solution for this problem exists with a practical edit of the existing equations. Replacing equations (5.1.8)-(5.1.11) with equations that represent a log mean temperature difference provides for a more robust set of system equations. Equations (4.4.4) and (4.4.5) in section (4.4.2) are similar to the concept being applied here. The main difference is that the heat exchanger in section (4.4.2) is only a heat exchanger so the log mean temperature difference is applied across the hot and cold side of the heat exchanger. The log mean temperature difference cannot be applied across the current heat exchanger because the thermoelectric barrier between the hot and cold side is removing energy from the system. Therefore, the log mean temperature difference is applied to each side of the heat exchanger and related to the temperatures at the thermoelectric module surface, T_h and T_c. This is seen in the following equations:

$$\Delta T_{lm,h} = \frac{T_{h,i} - T_{h,o}}{ln\left(\frac{T_{h,i} - T_h}{T_{h,o} - T_h}\right)} \tag{5.1.12}$$

and

$$\Delta T_{lm,h} = \frac{T_{c,i} - T_{c,o}}{ln\left(\frac{T_c - T_{c,o}}{T_c - T_{c,i}}\right)} \quad (5.1.13)$$

where $\Delta T_{lm,h}$ is the hot side log mean temperature difference and $\Delta T_{lm,c}$ is the cold side log mean temperature difference. Use of these equations relies on the assumptions that axial conduction is negligible, potential and kinetic energy changes are negligible, fluid specific heats are constant, the overall heat transfer coefficient is constant, and the heat exchanger is insulated from its surroundings [37]. The entire heat exchanger being modeled as insulated from its surroundings holds true at this stage but is no longer true in the final set of equations. This is acceptable because the lost heat is represented by a parallel heat flow path to the equation that factors in log mean temperature difference which can be seen in equations (5.1.16) and (5.1.18). Also, if the hot or cold side of the heat exchanger is isolated from the system, the log mean temperature difference for that side is as if it is "insulated" from its surroundings.

Equations (5.1.12) and (5.1.13) need to be related back to the system to utilize their value so equations (5.1.10) and (5.1.11) need to be replaced. Equation (5.1.10) represents the total heat rate entering the hot side of the thermoelectric module and the hot side of the insulation. The change in temperature can now be represented by $\Delta T_{lm,h}$ and the thermal resistance is captured in an UA-value.

$$UA_h \Delta T_{lm,h} = q_h + q_{ins} \quad (5.1.14)$$

UA_h represents the hot side UA-value which consists of the thermal conductance through the face of the hot side of the heat exchanger, the thermal conductance through the finned compact heat exchangers on the hot side, the convection coefficient of the hot side fluid flow, and any other piece of the system on the hot side between T_h, $T_{h,i}$, and $T_{c,o}$.

A similar equation exists for the cold side of the heat exchanger where the total heat rate leaving the thermoelectric module and the insulation is accounted for.

$$UA_c \Delta T_{lm,c} = q_c + q_{ins} \quad (5.1.15)$$

UA_c represents the cold side UA-value which consists of the thermal conductance through the face of the cold side of the heat exchanger, the thermal conductance through the finned compact heat exchangers on the cold side, the convection coefficient of the cold side fluid flow, and any other piece of the system on the cold side between T_c, $T_{c,i}$, and $T_{c,o}$. Determining a values for UA_h and UA_c will be discussed in the subsections of section 5.3.

The system of equations is now more robust but still lacking some analysis that encompasses real world phenomena. Heat losses to the surrounding environment require attention to complete the counter flow thermoelectric heat exchanger system of equations.

5.1.4 Factoring in Heat Losses to the Environment

To use this model in a practical application requires the modeling of the effects of the surrounding environment. Many other models neglect to incorporate this analysis limiting the accuracy of their solution. An actual heat exchanger will inevitably lose heat energy to its surrounding environment regardless of how well insulated it is. Insulation will minimize environmental losses and this portion of the model has the ability to determine the heat lost.

Currently, the system model consists of equations (5.1.1), (5.1.3)-(5.1.7), and (5.1.12)-(5.1.15) with ten unknowns. This setup provides a solvable system and a powerful tool for theoretical power prediction. To improve the system model and provide a realistic scenario, heat lost to the environment needs to accounted for. Each side of the heat exchanger has its lost heat energy factored in with the variables $q_{losses,h}$ and $q_{losses,c}$ where $q_{losses,h}$ is the hot side of the heat exchanger's lost heat energy and $q_{losses,c}$ is the cold side of the heat exchanger's lost heat energy. Figure 5.6 shows the lost energy accounted for and the LMTD change to the model.

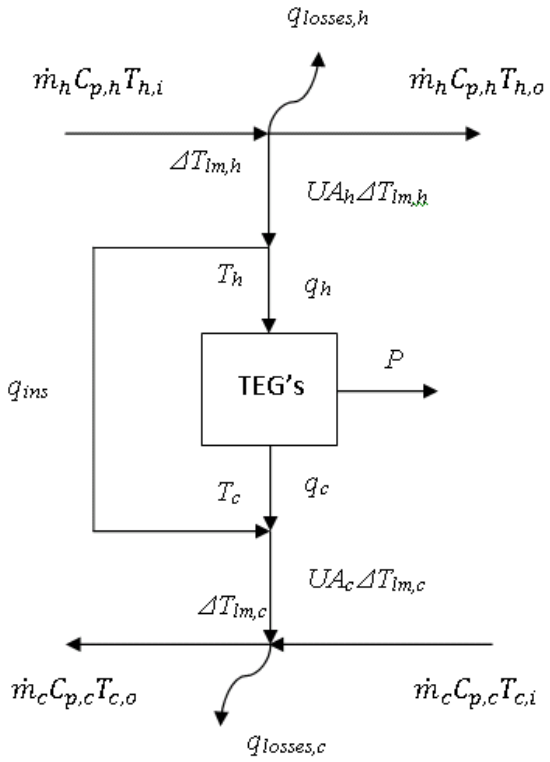


Figure 5.6: Energy balance circuit of final model for a zone.

Equations (5.1.6) and (5.1.7) represent the total heat energy in the hot and cold sides of the heat exchanger, respectively. When accounting for heat losses, equation (5.1.6) develops into equation (5.1.16).

$$\dot{m}_h C_{p,h}\left(T_{h,i} - T_{h,o}\right) = UA_h \Delta T_{lm,h} + q_{losses,h} \quad (5.1.16)$$

This equation accurately depicts the heat transfer in the hot side of the heat exchanger. To help with the explanation, the term on the left side of the equation will be broken up and rearranged resulting in equation (5.1.17).

$$\dot{m}_h C_{p,h} T_{h,i} = UA_h \Delta T_{lm,h} + q_{losses,h} + \dot{m}_h C_{p,h} T_{h,o} \quad (5.1.17)$$

Equation (5.1.17) is a balance equation where the energy into the system equals the energy out of the system. The term on the left hand side of this equation is the total energy entering the hot side of the heat exchanger. The right side of the equation represents the three paths that the energy is constrained to. The first term on the right side of the equation represents the energy that is distributed to the thermoelectric modules. The second term is the heat energy lost through the shell of the hot side of the heat exchanger out into the environment. The determination of this variable is explained later in this section. The third term on the right side of equation (5.1.17) is all of the energy that leaves the hot side of the heat exchanger and isn't used by the modules or lost to the environment. This is all relatively intuitive in this instance whereas it isn't quite as transparent for the heat lost from the cold side of the heat exchanger.

The losses from the cold side of the heat exchanger are accounted for in a similar manner to equation (5.1.16) in equation (5.1.18).

$$\dot{m}_c C_{p,c}\left(T_{c,o} - T_{c,i}\right) = UA_c \Delta T_{lm,c} - q_{losses,c} \quad (5.1.18)$$

In this form of the equation, it is less intuitive as to why $q_{losses,c}$ is subtracted from the right side of the equation when $q_{losses,h}$ is added to the right side of equation (5.1.16). Rearranging equation (5.1.18) will provide the answer to this.

$$\dot{m}_c C_{p,c} T_{c,o} + q_{losses,c} = UA_c \Delta T_{lm,c} + \dot{m}_c C_{p,c} T_{c,i} \quad (5.1.19)$$

In equation (5.1.19), the left side of the equation represents the energy leaving the cold side of the heat exchanger. The energy leaving the cold side is equal to the energy entering the cold side which is the terms on the right hand side. The first term on the right hand side of the equation is the energy that enters the flow stream that has made it through the thermoelectric modules and not been converted into electrical energy. The second term on the right hand side is the energy that enters the cold side of the heat exchanger in the flow stream. Both the terms on the left hand side represent energy leaving the heat exchanger. The first term on the left is the energy that leaves the heat exchanger through the fluid flow. The second term is the energy lost through the shell of the cold side of the heat exchanger.

Both $q_{losses,h}$ and $q_{losses,c}$ need to be determined based on the inlet and outlet temperatures of their respective sides of the heat exchanger. Equations (5.1.20) and (5.1.21) show how $q_{losses,h}$ and $q_{losses,c}$ are determined, respectively.

$$q_{losses,h} = UA_{h,envir} \frac{T_{h,i} - T_{h,o}}{ln\left(\frac{T_{h,i} - T_{\infty}}{T_{h,o} - T_{\infty}}\right)} \quad (5.1.20)$$

$$q_{losses,c} = UA_{c,envir} \frac{T_{c,o} - T_{c,i}}{ln\left(\frac{T_{c,o} - T_{\infty}}{T_{c,i} - T_{\infty}}\right)} \quad (5.1.21)$$

The newly introduced parameters are T_{∞} which represents the ambient temperature of the heat exchanger and is considered a known property, $UA_{h,envir}$ which is the thermal conductance for the hot side heat exchanger shell, and $UA_{c,envir}$ which is the thermal conductance for the cold side heat exchanger shell. Each of these equations utilizes a log mean temperature difference between the inlets and outlets of the flow related to the ambient temperature. Each of the UA-values is considered a known property as the parameters that factor into it are part of the heat exchanger design. The thickness of the shell and the conductive coefficient of the heat exchanger material are needed to assist in finding the resistance due to conduction. Shell surface area is determined by heat exchanger parameters that are input into the design parameters. Building the heat exchanger based on input parameters will be discussed in section 5.3. The other parameter that is needed to determine the UA-values is a U-value representative of the natural convection surrounding the heat exchanger.

Now that the environmental losses are accounted for, the number of equations and unknowns in the system has extended to twelve. The system consists of (5.1.1), (5.1.3)-(5.1.5), (5.1.12)-(5.1.16), (5.1.18), and (5.1.20)-(5.1.21). The twelve unknowns are P, q_h, q_c, T_h, T_c, q_{ins}, $\Delta T_{lm,h}$, $\Delta T_{lm,c}$, $T_{h,o}$, $T_{c,i}$, $q_{losses,h}$, and $q_{losses,c}$. A counter flow heat exchanger with thermoelectric generators can be solved for but these fundamental equations can be expanded to work for other types of heat exchangers. If one side of the heat exchanger is known to be isothermal or has a constant heat rate, the model developed can be arranged to work for the proposed heat exchanger.

5.1.5 Isothermal and Constant Heat Rate Heat Exchangers

The twelve equations that have been defined to model the counter flow heat exchanged can be manipulated and edited to solve for an isothermal or constant heat rate heat exchanger. The isothermal heat exchanger assumes one side of the heat exchanger to be of the same temperature across that side and remain that temperature at steady state. In this case, the isothermal side is defined as the cold side of the heat exchanger. The constant heat rate heat exchanger is defined very similar to the isothermal heat exchanger. The main difference is that the constant heat rate is applied across one side of the heat exchanger as if there was a heating or cooling pad that remains constant at steady state. Even though it would seem more practical to consider a constant heat rate source on the hot side, the cold side will have the constant heat rate to remain consistent with the isothermal cold side.

Many of the counter flow heat exchanger equations can be used for the constant heat rate and isothermal heat exchanger analysis. Taking the equations from the previous analysis and eliminating the ones that represent cold side fluid flow (5.1.13), (5.1.15), and (5.1.18), leaves (5.1.1), (5.1.3)-(5.1.5), (5.1.12), (5.1.14), (5.1.16), and (5.1.20)-(5.1.21). Also, equation (5.1.21) can be eliminated because cold side losses are no longer considered part of the system as the side remains constantly isothermal or at the same heat rate. Losses will not affect the system performance. A breakdown in the system analysis on the cold side of the thermoelectric modules is apparent. A new variable and equation (5.1.22) need to be introduced to make the system consistent.

$$q_{c,tot} = q_c + q_{ins} \tag{5.1.22}$$

Equation (5.1.22) actually replaces (5.1.15) where $q_{c,tot}$ is the total heat rate for the cold side of the heat exchanger. This is representative of the total energy of the cold side whether it is an isothermal or a constant heat rate heat exchanger. This is also needed because q_c is only the cold side heat rate for each module or group of modules for each zone. The insulation heat rate, q_{ins}, also exists on the cold side of the heat exchanger so a new variable must exist to capture that.

When the heat exchanger is considered to have an isothermal face on the cold side, T_c is considered to be a known parameter. This means that $q_{c,tot}$ is an unknown variable and the entire set of unknowns is P, q_h, q_c, T_h, q_{ins}, $\Delta T_{lm,h}$, $T_{h,o}$, $q_{losses,h}$, and $q_{c,tot}$. The nine equations used to solve these variables are (5.1.1), (5.1.3)-(5.1.5), (5.1.12), (5.1.14), (5.1.16), (5.1.20), and (5.1.22).

When the heat exchanger is considered to have a constant heat rate on the cold side, $q_{c,tot}$ is considered to be a known parameter. This means that T_c is an unknown variable and the entire set of unknowns is P, q_h, q_c, T_h, q_{ins}, $\Delta T_{lm,h}$, $T_{h,o}$, $q_{losses,h}$, and T_c. The same nine equations are used to solve these variables which are a powerful tool when it comes to the programming portion discussed in Chapter 6. It provides for versatility and less redundancy in the code.

Now that the system equations have been developed for the model, it is necessary to determine parameters that are considered known. The next section discusses the several options of methods used to final the thermoelectric module parameters. These include the Seebeck coefficient (α), the module thermal resistance (R_{th}), and the module electrical resistance (R_e) which can be seen in equations (5.1.3) and (5.1.4).

5.2 Determining Thermoelectric Generator Key Parameters

Each thermoelectric generator's performance can be determined from three key parameters. These parameters include the Seebeck coefficient (α), the module thermal resistance (R_{th}), and the module electrical resistance (R_e). The Seebeck coefficient defines the phenomenon of current

in the semiconductors as a result of temperature difference across the module. The module thermal resistance is the resistance across the two faces of the thermoelectric wafer and the electrical resistance is the opposition of the flow of electric current through the semiconductors inside the module. Knowing these three parameters, the operating temperatures on either side of the module, and the load resistance allows for the prediction of power output by equations (5.1.1), (5.1.3), and (5.1.4).

It is important to note that the next several sub-sections provide different methods for finding the key parameters. Each of these methods is considered an option in the TEPSS code and can be utilized with the right inputs to the calculation. Options include predetermined parameters by the user through empirical means, parameters determined through the input material properties and geometries of the thermoelectric legs and ceramic wafers, two different ways to find the parameters using thermoelectric cooler manufacturer provided data, and the use of thermoelectric generator manufacturer data.

5.2.1 Predetermined Key Parameters through Empirical Means

Any user of TEPSS has the ability to directly supply the aforementioned parameters. This option would typically be used if the modules were experimentally tested in a separate system similar to the one shown in Figure 3.3. The parameters can be derived from the recorded data using the appropriate equations.

For the TEPSS code, parameters are input for an individual module. These parameters need to be expanded for all of the modules in the zone. The TEPSS user has the ability to define the number of modules in a zone indirectly. It is possible to input the number of modules aligned in series by flow path meaning the module is downstream of the previous module. It is also possible to input the number of modules aligned in parallel by flow path meaning the module is next to another module as the flow progresses as seen in Figure 5.7.

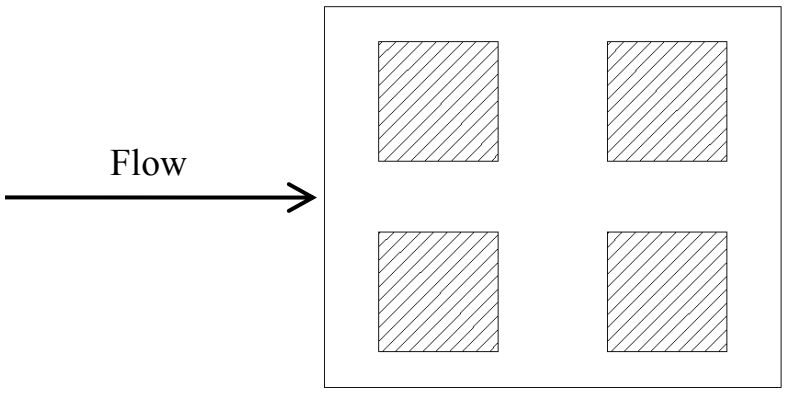


Figure 5.7: Grid of modules in a thermoelectric power unit zone.

This creates a grid of modules which means that the number in parallel ($N_{mod,par}$) and the number in series ($N_{mod,ser}$) multiplied together yields the total number of modules in a zone ($N_{mod,zone}$) which can be seen in equation (5.2.1).

$$N_{mod,zone} = N_{mod,par}N_{mod,ser} \quad (5.2.1)$$

All of the modules in a zone are considered to be electrically in series which means that the total Seebeck coefficient for a group of modules and the total electrical resistance for a group of modules can be seen in equations (5.2.2) and (5.2.3), respectively.

$$\alpha = \alpha_{single}N_{mod,zone} \quad (5.2.2)$$

$$R_e = R_{e,single}N_{mod,zone} \quad (5.2.3)$$

α_{single} is the Seebeck coefficient for a single module and $R_{e,single}$ is the internal electrical resistance for a single module. The parameters used in equations (5.1.3) and (5.1.4) are represented as the total Seebeck coefficient and total internal electrical resistance for the zone. Because all of the system equations are used to solve a zone, the thermoelectric generator key parameters are needed at the zone level.

By the nature of the heat exchanger and module placement, the modules are thermally in parallel which is seen in equation (5.2.4). This means that the more modules that are considered to make up a zone, the less the thermal resistance will be. It also means that the Seebeck coefficient and internal electrical resistance will increase.

$$R_{th} = \frac{R_{th,single}}{N_{mod,zone}} \quad (5.2.4)$$

$R_{th,single}$ is the thermal resistance for a single module which also needs to be expanded to represent a zone. The same applies for $R_{th,single}$ as it does for α_{single} and $R_{e,single}$ when it comes to the use of equations (5.1.3) and (5.1.4).

A couple other parameters are needed for the thermoelectric generators to determine the design of the heat exchanger. These include the thickness of the module, t_{mod}, and the width of the module, w_{mod}. The reason the length of the module is not needed is because thermoelectric modules are often square in shape. This is assumed throughout the analysis for simplicity of design. Thermoelectric modules can be rectangular or vary in shape depending on the manufacturers design. The module width and thickness are input by the TEPSS user for all options for determining the key parameters of the TEG besides the next option where the thickness and width need to be calculated.

5.2.2 Key Parameters Calculated from P-type and N-type Semiconductor Leg Geometries and Material Properties

The ability to develop a thermoelectric module based on input parameters and calculate the modules performance is of great significance to TEPSS. More importantly, the ability to give different parameters for the P-type and N-type semiconductor legs and find the key parameters is a useful asset. To develop these equations, a single thermoelectric pair needs to be analyzed using methods from Sandoz-Rosado [15]. Figure 2.1 represents the leg pair to be analyzed where *x* is the distance from the hot surface of the leg pair. The analysis is started with an energy balance for each leg at the surface where *x* equals zero.

$$q_p = (\alpha_p - \alpha^*) I T_{HE} - \lambda_p A_p \frac{dT_p}{dx} \tag{5.2.5}$$

$$q_n = -(\alpha_n - \alpha^*) I T_{HE} - \lambda_n A_n \frac{dT_n}{dx} \tag{5.2.6}$$

Subscripts *p* and *n* represent the positive and negative semiconductor, respectively. q is the heat flow into a leg, α is the Seebeck coefficient for a leg, α* is a relative Seebeck coefficient, I is the current in the leg pair which can flow either way, T_{HE} is the hot side temperature of the leg pair at the leg pair surface, λ is the thermal conductance of a leg, A is the cross sectional area of a leg, and dT/dx is the change in temperature over the change in length for a leg. To remove the differential in equations (5.2.5) and (5.2.6), the heat diffusion equation for each leg needs to be used.

$$-\lambda_p A_p \frac{d^2T}{dx^2} = \frac{I^2 \rho_p}{A_p} \tag{5.2.7}$$

$$-\lambda_n A_n \frac{d^2T}{dx^2} = \frac{I^2 \rho_n}{A_n} \tag{5.2.8}$$

ρ is the electrical resistivity of each leg of the semiconductor pair. Each of these second derivatives has two initial conditions which makes them solvable. The initial conditions are the temperature at *x* equals zero is T_{HE} for each leg, the temperature at *x* equals the length of the P-type leg, L_p, is T_{CE}, and the temperature at *x* equals the length of the N-leg, L_n, is T_{CE}. T_{CE} is defined as the cold side temperature of the leg pair at the leg pair surface. Even though the legs could be different lengths, they are assumed to be the same temperature at their ends. The initial conditions are shown in equations (5.2.9) and (5.2.10).

$$T(x = 0) = T_{HE} \tag{5.2.9}$$

$$T(x = L_p, x = L_n) = T_{CE} \tag{5.2.10}$$

Solving the differential equations using the given initial conditions yields equations (5.2.11) and (5.2.12).

$$T_p = -\frac{I^2\rho_p}{\lambda_p A_p^2}\frac{x^2}{2} + \left(\frac{T_{CE}-T_{HE}}{L_p} + \frac{I^2\rho_p L_p}{2\lambda_p A_p^2}\right)x + T_{HE} \tag{5.2.11}$$

$$T_p = -\frac{I^2\rho_n}{\lambda_n A_n^2}\frac{x^2}{2} + \left(\frac{T_{CE}-T_{HE}}{L_n} + \frac{I^2\rho_n L_n}{2\lambda_n A_n^2}\right)x + T_{HE} \tag{5.2.12}$$

The next steps of the analysis are identical for both the P-type and N-type legs so the analysis will only be shown for the P-type leg. To utilize equations (5.2.11) and (5.2.12), the derivative needs to be taken to then substitute the answers into equations (5.2.5) and (5.2.6).

$$\frac{dT_p}{dx} = -\frac{I^2\rho_p}{\lambda_p A_p^2}x + \frac{T_{CE}-T_{HE}}{L_p} + \frac{I^2\rho_p L_p}{2\lambda_p A_p^2} \tag{5.2.13}$$

Rearranging equation (5.2.13) into a useable form provides equation (5.2.14).

$$\lambda_p A_p \frac{dT_p}{dx} = -\frac{I^2\rho_p}{A_p}\left(x - \frac{L_p}{2}\right) + \frac{\lambda_p A_p (T_{CE}-T_{HE})}{L_p} \tag{5.2.14}$$

The substitution of equation (5.2.14) into equation (5.2.5) with x set equal to zero will yield equation (5.2.15). x is set equal to zero because it will allow q_h to be found.

$$q_p(x=0) = (\alpha_p - \alpha^*)IT_{HE} - \frac{\lambda_p A_p (T_{CE}-T_{HE})}{L_p} - \frac{I^2\rho_p L_p}{2A_p} \tag{5.2.15}$$

As was stated, the analysis for the N-type leg was the same and yields similar results as seen in equation (5.2.16).

$$q_n(x=0) = -(\alpha_n - \alpha^*)IT_{HE} - \frac{\lambda_n A_n (T_{CE}-T_{HE})}{L_n} - \frac{I^2\rho_n L_n}{2A_n} \tag{5.2.16}$$

Now that q_p and q_n have been found for the hot side of the leg pair, it is possible to solve for q_h by adding equations (5.2.15) and (5.2.16) together. The result is equation (5.2.17).

$$q_h = (\alpha_p - \alpha_n)IT_{HE} + (T_{HE} - T_{CE})\left[\frac{\lambda_p A_p}{L_p} + \frac{\lambda_n A_n}{L_n}\right] - \frac{I^2}{2}\left[\frac{\rho_p L_p}{A_p} + \frac{\rho_n L_n}{A_n}\right] \tag{5.2.17}$$

This equation is for one leg pair and needs to be expanded to represent multiple pairs represented by N. This resembles the standard thermoelectric equation as seen in equation (1.1.3).

$$q_h = N(\alpha_p - \alpha_n)IT_{HE} + (T_{HE} - T_{CE})N\left[\frac{\lambda_p A_p}{L_p} + \frac{\lambda_n A_n}{L_n}\right] - \frac{I^2}{2}N\left[\frac{\rho_p L_p}{A_p} + \frac{\rho_n L_n}{A_n}\right] \tag{5.2.18}$$

The first term on the right hand side of equation (5.2.18) can be separated and turned into a new variable to represent the Seebeck effect for the module, $\alpha_{p,n}$. This is the same as the usual representation of the Seebeck coefficient which can be seen in equation (5.2.19). Each of the variables on the right hand side of the equation are user given and considered known properties. These variables are considered design inputs.

$$\alpha_{p,n} = N(\alpha_p - \alpha_n) \tag{5.2.19}$$

The second term on the right hand side of equation (5.2.18) can be factored out and is representative of the thermal conductance, K_{th}. Taking the inverse of K_{th} produces the thermal resistance, R_{th}, which is shown in equation (5.2.20). Once again, each of the variables on the right hand side of the equation is considered a known property.

$$R_{th} = \frac{1}{\frac{N\lambda_p A_p}{L_p} + \frac{N\lambda_n A_n}{L_n}} \tag{5.2.20}$$

This thermal resistance is used for all of the leg pairs in a module but does not account for the thermal resistance of the surrounding ceramic, R_{cer}. Additional analysis is needed to find R_{cer}. Figure 1.1 shows the existence of the ceramic layers surrounding the leg pair.

Each of these ceramic layers uses a simple thermal conductive resistance to find a suitable equation. The thickness of the layer is considered to be L_{cer} and the material conductivity is k_{cer}. The legs are thermally in parallel so the ceramic thermal resistance is divided by the number of leg pairs. The most difficult part in determining the ceramic resistance is the surface area to use. When considering the area of both of the semiconductor legs, equation (5.2.21) is produced.

$$R_{cer} = \frac{L_{cer}}{Nk_{cer}}\left(\frac{1}{A_p + A_n}\right) \tag{5.2.21}$$

However, this is not entirely accurate. Part of the area of the ceramic is not considered in this analysis meaning a method is needed to find the actual thermal resistance of the whole piece of ceramic. Multiplying the denominator of equation (5.2.21) by an area ratio, A_{ratio}, representing the ratio of the surface area of the ceramic to the surface area of the combined leg pairs yields equation (5.2.22). This equation does not account for any spreading resistance that may occur. All of the variables on the right hand side of the equation are known parameters which is the same as before.

$$R_{cer} = \frac{L_{cer}}{Nk_{cer}}\left(\frac{1}{(A_p + A_n)A_{ratio}}\right) \tag{5.2.22}$$

This equation now represents one side of the thermoelectric module ceramic thermal resistance. Because there are typically two pieces of ceramic and these pieces are thermally in series, equation (5.2.22) can be multiplied by two to represent the total ceramic thermal resistance of a module. However, when this is used in the system of equations, each piece of ceramic is used independent of each other. They are assumed to be the same as that is typical of most manufactured thermoelectric modules.

It should also be noted that equation (5.2.22) is only for one ceramic piece of one module. To find the total thermal resistance of either the hot side or cold side of all the modules in a zone, an equation similar to (5.2.4) is needed. Equation (5.2.23) is used to find the thermal resistance of

all the modules in the zone as they are in parallel. This is only representative of one side of the modules though. Each side of the module is taken into account later in this section as the system equations are changed to reflect this method of finding the key parameters.

$$R_{cer} = \frac{R_{cer,single}}{N_{mod,zone}} \tag{5.2.23}$$

Now that A_{ratio}, the ratio of the surface area of the ceramic to the surface area of the combined leg pairs, has been introduced, it is possible to develop a method for determining the width of a module. Equation (5.2.24) is developed for this purpose. It is assumed that the module is square like each of the other method for finding the thermoelectric key parameters. It takes into account the cross sectional area of each of the legs and multiples that by the number of pairs and the A_{ratio}. This provides the total face area of the ceramic.

$$w = \sqrt{N\left(A_p + A_n\right)A_{ratio}} \tag{5.2.24}$$

Because the module is currently being sized up, it is best to consider how the thickness of the module is to be found. The thickness is the length of the longest leg in the pair plus two times the ceramic thickness, one for each side of the module. This can be seen in equation (5.2.25). If the legs of the module are not the same length, a spacer or bridge is typically used between the ceramic and the leg. This is not modeled and does not have any electrical impact on the module. It could have a thermal impact but is assumed to not and is considered to have the same thermal properties of the shorter leg.

$$t = \max\left(L_p, L_n\right) + 2L_{cer} \tag{5.2.25}$$

The third term on the right hand side of equation (5.2.18) can be used to find the electrical resistance of all the leg pairs in the module. It factors out of equation (5.2.18) into equation (5.2.26).

$$R_e = \frac{N\rho_p L_p}{A_p} + \frac{N\rho_n L_n}{A_n} \tag{5.2.26}$$

This equation represents the electrical resistance of the leg pairs but does not account for the electrical resistance of the joints between each leg and the surrounding leg pairs. For each leg pair, four interfaces are present, one on each connecting face of each leg. This means that there are two electrical contact resistances for each P-type leg area and two electrical contact resistances for each N-type leg area. The electrical contact resistance is denoted by $R''_{e,con}$ and is given in units that are equivalent to electrical resistance times unit area. Because the electrical components are all in series, each of the components of the equation can be added together which is seen in equation (5.2.27). The variables that make up the internal electrical resistance are known design parameters that are predefined.

$$R_e = \frac{N\rho_p L_p}{A_p} + \frac{N\rho_n L_n}{A_n} + \frac{2NR''_{e,con}}{A_p} + \frac{2NR''_{e,con}}{A_n} \quad (5.2.27)$$

Each of the key thermoelectric parameters is now considered to be known and is ready to be used for the system analysis. There is one problem with the final set of system equations developed in section 5.1. Each of those equations treats the hot side thermoelectric temperature and cold side thermoelectric temperature to be at the outer surfaces of the ceramic wafers. The temperature for the equations that were described in this section require the hot side and cold side thermoelectric temperatures to be at the inner surfaces of the ceramic wafers or actually the temperature at the surfaces of the legs. This means that the system equations developed in section 5.1 will need to be edited to reflect this difference in temperature to accurately provide results.

For a counter flow heat exchanger, the system equations are (5.1.1), (5.1.3)-(5.1.5), (5.1.12)-(5.1.16), (5.1.18), and (5.1.20)-(5.1.21)with twelve unknowns which are P, q_h, q_c, T_h, T_c, q_{ins}, $\Delta T_{lm,h}$, $\Delta T_{lm,c}$, $T_{h,o}$, $T_{c,i}$, $q_{losses,h}$, and $q_{losses,c}$. For a heat exchanger with an isothermal face on the cold side or a constant heat rate on the cold side, the system of equations are made up by (5.1.1), (5.1.3)-(5.1.5), (5.1.12), (5.1.14), (5.1.16), (5.1.20), and (5.1.22) with nine unknown variables. These are considered to be P, q_h, q_c, T_h, q_{ins}, $\Delta T_{lm,h}$, $T_{h,o}$, $q_{losses,h}$, and $q_{c,tot}$ for the isothermal side heat exchanger. When the heat exchanger has a constant heat rate side, the nine unknown variables are P, q_h, q_c, T_h, q_{ins}, $\Delta T_{lm,h}$, $T_{h,o}$, $q_{losses,h}$, and T_c.

Equations (1.1.3) and (1.1.4) for q_h and q_c, respectively, can be modified to represent the temperatures at the junctions of the leg pairs as opposed to the outer surfaces of the ceramic wafers. Equations (5.2.28) and (5.2.29) will provide a fix to the problem identified.

$$q_h = \alpha_{p,n} I T_{HE} + \frac{T_{HE} - T_{CE}}{R_{th}} - \frac{1}{2} R_e I^2 \quad (5.2.28)$$

$$q_c = \alpha_{p,n} I T_{CE} + \frac{T_{HE} - T_{CE}}{R_{th}} + \frac{1}{2} R_e I^2 \quad (5.2.29)$$

To eliminate a variable, an equation for current, I, is needed. This equation is similar to equation (5.1.2) with the main difference that the current is only represented through the leg pairs as it should be. If (5.1.2) was used, it would suggest that there was current flowing through the ceramic which is not true because the ceramic is an insulator. The new equation for current can be seen in (5.2.30).

$$I = \frac{\alpha_{p,n}(T_{HE} - T_{CE})}{R_e + R_L} \quad (5.2.30)$$

Substituting (5.2.30) into (5.2.28) and (5.2.29) creates equations suitable for replacing equations (5.1.3) and (5.1.4), respectively. Equations (5.2.31) and (5.2.32) can now be used in the system model.

$$q_h = \alpha_{p,n}\left[\frac{\alpha_{p,n}(T_{HE}-T_{CE})}{R_e+R_L}\right]T_{HE} + \frac{T_{HE}-T_{CE}}{R_{th}} - \frac{1}{2}R_e\left[\frac{\alpha_{p,n}(T_{HE}-T_{CE})}{R_e+R_L}\right]^2 \quad (5.2.31)$$

$$q_c = \alpha_{p,n}\left[\frac{\alpha_{p,n}(T_{HE}-T_{CE})}{R_e+R_L}\right]T_{CE} + \frac{T_{HE}-T_{CE}}{R_{th}} + \frac{1}{2}R_e\left[\frac{\alpha_{p,n}(T_{HE}-T_{CE})}{R_e+R_L}\right]^2 \quad (5.2.32)$$

Equations (5.2.31) and (5.2.32) can be used with equations (5.2.19), (5.2.20), and (5.2.27) to associate the thermoelectric key parameters with the system model. However, the temperatures, T_{HE} and T_{CE}, do not relate back to the entire system model. They are capturing the temperatures at the ends of the legs but fail to be directly affected by the temperatures of the heat exchanger flow. The ceramic material is not modeled in this system at this point. Additional equations are needed to relate each physical object to the model. The direct relation that is needed is the one between the leg temperature and the module surface temperature for each side of the module. The ceramic material is the temperature drop between these locations and the heat rate through the ceramic is the same as the heat rate into the module. The same effects are true for the cold side of the module as well. Equations (5.2.33) and (5.2.34) show this.

$$q_h = \frac{T_h - T_{HE}}{R_{cer}} \quad (5.2.33)$$

$$q_c = \frac{T_{CE} - T_c}{R_{cer}} \quad (5.2.34)$$

Adding these equations to the system makes it solvable again. Two new unknowns, T_{HE} and T_{CE}, and two new equations, (5.2.33) and (5.2.34), were introduced to the system model. Equations (5.1.3) and (5.1.4) were edited to reflect the changes made to the system model which introduced equations (5.2.31) and (5.2.32). When using key parameters calculated from P-type and N-type semiconductor leg geometries and material properties, the system equations for a counter flow heat exchanger are (5.1.1), (5.1.5), (5.1.12)-(5.1.16), (5.1.18), (5.1.20)-(5.1.21), and (5.2.31)-(5.2.34) with fourteen unknowns which are P, q_h, q_c, T_h, T_c, q_{ins}, $\Delta T_{lm,h}$, $\Delta T_{lm,c}$, $T_{h,o}$, $T_{c,i}$, $q_{losses,h}$, $q_{losses,c}$, T_{HE}, and T_{CE}. When using key parameters calculated from P-type and N-type semiconductor leg geometries and material properties, the system equations for a heat exchanger with an isothermal face on the cold side or a constant heat rate on the cold side are made up by (5.1.1), (5.1.5), (5.1.12), (5.1.14), (5.1.16), (5.1.20), (5.1.22), and (5.2.31)-(5.2.34) with eleven unknown variables. These are considered to be P, q_h, q_c, T_h, q_{ins}, $\Delta T_{lm,h}$, $T_{h,o}$, $q_{losses,h}$, $q_{c,tot}$, T_{HE}, and T_{CE} for the isothermal side heat exchanger. When the heat exchanger has a constant heat rate side, the nine unknown variables are P, q_h, q_c, T_h, q_{ins}, $\Delta T_{lm,h}$, $T_{h,o}$, $q_{losses,h}$, T_c, T_{HE}, and T_{CE}.

5.2.3 Method I for Finding Key Parameters from Thermoelectric Cooler Manufacturer Provided Data

Many thermoelectric modules are sold by manufacturers as thermoelectric coolers (TECs). Several of the common applications for the devices are cooling purposes where electricity is

applied to the circuit inducing a current through the module which then heats one side of the device and cools the other. The key parameters such as the Seebeck effect, internal electrical resistance, and thermal resistance are not reported. However, other design criteria are reported by the manufacturer which indirectly describes the performance of the module as a thermoelectric generator. Zhaoxia Luo [40] reports a couple of methods for finding the key parameters based on information that a vendor would provide.

The first method for finding the key parameters uses maximum temperature difference (ΔT_{max}), hot side module temperature (T_h), maximum current (I_{max}), and maximum voltage (V_{max}). All of these variables are typically provided in the manufacturer data sheets. It should be noted that the temperature of the TEC hot face is typically held at a specific temperature such as 25°C when obtaining these parameters. I_{max} is the input current resulting in the greatest ΔT_{max} when q_c is equal to zero. V_{max} is the measured voltage at this condition. With this data from the manufacturer, the thermoelectric key parameters can be obtained. The Seebeck coefficient can be seen in equation (5.2.35).

$$\alpha = \frac{V_{max}}{T_h} \tag{5.2.35}$$

The thermal resistance of the module is obtained using equation (5.2.36).

$$R_{th} = \frac{2T_h \Delta T_{max}}{(T_h - \Delta T_{max}) V_{max} I_{max}} \tag{5.2.36}$$

And the internal electrical resistance of the module based on TEC manufacturer data is seen in equation (5.2.37).

$$R_e = \frac{(T_h - \Delta T_{max}) V_{max}}{T_h I_{max}} \tag{5.2.37}$$

Equations (5.2.35)-(5.2.37) are derived from the manipulations of basic thermoelectric equations. The derivations can be found in Zhaoxia Luo's article [40]. This article displays another set of equations that were derived from the same basic thermoelectric equations. They are shown in the next section.

5.2.4 Method II for Finding Key Parameters from Thermoelectric Cooler Manufacturer Provided Data

The second method used by Zhaoxia Luo [40] starts with the same basic thermoelectric equations. These equations are manipulated to find useful equations for finding the Seebeck coefficient, thermal resistance, and internal electrical resistance of the module. Thermoelectric cooler manufacturers also report performance data for the maximum amount of heat that can be absorbed at the cold face of the module, Q_{max}. Q_{max} occurs at the specific hot side temperature provided by the manufacturer, T_h, when the current equals I_{max} and ΔT equals 0°C. Using

manufacturer provided T_h, I_{max}, ΔT_{max}, and Q_{max}, the Seebeck coefficient can be found using equation (5.2.38).

$$\alpha = \frac{2Q_{max}}{I_{max}(T_h+\Delta T_{max})} \tag{5.2.38}$$

The thermal resistance of the module is obtained using the same manufacturer data points and inputting them into equation (5.2.39).

$$R_{th} = \frac{(T_h+\Delta T_{max})\Delta T_{max}}{(T_h-\Delta T_{max})Q_{max}} \tag{5.2.39}$$

To find the internal electrical resistance, it is required that the Seebeck coefficient and thermal resistance of the TEC be calculated using equations (5.2.38) and (5.2.39). This can be seen in equation (5.2.40).

$$R_e = \frac{\alpha^2 R_{th}(T_h-\Delta T_{max})^2}{2\Delta T_{max}} \tag{5.2.40}$$

Equations (5.2.38)-(5.2.40) are derived from the same manipulations of basic thermoelectric equations as were equations (5.2.35)-(5.2.27). The derivations can be found in Zhaoxia Luo's article [40]. These two methods for finding the key parameters are compared by Zhaoxia Luo for various data of TEC's obtained from manufacturers. As noted by Zhaoxia Luo, sometimes minor errors exist between the two calculation methods. One reason for this is that the equations used to find the parameters are ideal thermoelectric equations and the key parameters are treated as constants. The parameters actually vary with temperature which can provide a small amount of error. Another reason for the difference in results between methods is a result of the experimental data taken by the manufacturer. Each manufacturer uses different methods for finding these data points which can slightly skew the parameters to not match for each method [40].

5.2.5 Finding Key Parameters from Thermoelectric Generator Manufacturer Provided Data

Thermoelectric generator manufacturers provide different information in their datasheets than thermoelectric cooler manufacturers do. Even though the modules operate the same and are reversible, a TEG can be used as a TEC and vice versa, their reported data is considerably different. Open circuit voltage, V_{oc}, is recorded when the module is set up in conditions with a constant hot side temperature and a constant cold side temperature. These three reported data points can be used to find the Seebeck coefficient as seen in (5.2.41).

$$\alpha = \frac{V_{oc}}{T_h-T_c} \tag{5.2.41}$$

The internal electrical resistance is measured by the manufacturer and reported in the TEG datasheet. Since it is provided by the manufacturer, a calculation for its value is not required.

However, the reported electrical resistance value is needed to find the thermal resistance of the module. The calculated Seebeck coefficient is also needed to find the thermal resistance. Basic equation (1.1.3) is rearranged to utilize reported and calculated parameters. The module load current, I_{load}, is reported by the manufacturer for the testing temperatures. q_h is also reported in the datasheet for the same operating temperatures mentioned before. Using all of the reported and calculated values, the thermal resistance can be found using equation (5.2.42).

$$R_{th} = \frac{T_h - T_c}{q_h - \alpha I_{load} T_h + \frac{1}{2} I_{load}^2 R_e} \tag{5.2.42}$$

A method for finding the key parameters for a TEG using manufacturer provided data has been demonstrated. This method for finding necessary thermoelectric parameters as well as the others described in the previous sections, allows users of TEPSS to proceed forward with their thermoelectric power unit design. It is necessary to define the finned heat exchangers that are to be used in the power unit to determine thermal performance.

5.3 Finned Heat Exchangers

The overall heat exchanger model is in need of an increase in heat transfer effectiveness. To enable more heat to be channeled into the thermoelectric modules, finned heat exchangers are needed. The increase in surface area increases convection which in turn increases the heat transferred to or from the fluid depending on which side of the heat exchanger the fins are on. An appropriately selected material with high thermal conductivity is needed to maintain an effective level of conduction. These extended surfaces greatly improve the system model by increasing the value of UA_h in equation (5.1.16) and UA_c in equation (5.1.18). When more heat is transferred to the TEG's, more power is removed from the system and thus, generated by the heat exchanger.

Many fin options are available and selection of a configuration is dependent on several reasons. The overall intent of the finned heat exchangers is to increase surface area which will increase heat transfer. However, the increase in surface area will increase the pressure drop of the flow of fluid passing over the heat exchangers. This trade-off needs to be balanced as part of the design of the fin assemblies. The design also needs to reflect material and manufacturing costs but this is taken into account in section 6.3. The fin options to be discussed and that are available for TEPSS include rectangular straight fins, offset strip fins, and pin fins which have the option of either being aligned or staggered.

Before the extended surfaces can be modeled, the size of the heat exchanger unit must be determined. Sizing up the heat exchanger is based on the size, orientation, and number of modules. Instead of finding the whole heat exchanger size, the size of an individual zone is found and then extended to represent the entire heat exchanger. First, the length and width of all the modules combined within a zone is determined. Because the modules are assumed to be square

as they often are, length and width differ by the number of modules defined by flow orientation. From section (5.2), $N_{mod,par}$ and $N_{mod,ser}$ exist to aid in developing the orientation of the modules in a zone. This means that the total length of the modules in the zone can be demonstrated by equation (5.3.1) and the total width of the modules is seen in equation (5.3.2).

$$L_{mod,zone} = w_{mod} N_{mod,ser} \tag{5.3.1}$$

$$w_{mod,zone} = w_{mod} N_{mod,par} \tag{5.3.2}$$

$L_{mod,zone}$ is the length of all of the modules in a zone if they were directly adjacent to each other. $w_{mod,zone}$ is the width of all the modules in a zone if they were side by side. These parameters can be used to develop two more equations and important values. One of these values is the surface area of the modules in the zone. The other value is a ratio of length to width. Each of these can be used to determine the dimensions of the zone which are of more value and a necessity in determining performance of the finned heat exchangers. Equation (5.3.3) shows the surface area of the modules and equation (5.3.4) shows the ratio of length to width.

$$A_{mod,zone} = L_{mod,zone} w_{mod,zone} \tag{5.3.3}$$

$$\beta_{lw} = \frac{L_{mod,zone}}{w_{mod,zone}} \tag{5.3.4}$$

$A_{mod,zone}$ is the surface area of all the modules in the zone and β_{lw} is the ratio of the length of all the modules in the zone to the width of all the modules in the zone. These two parameters, in addition to a user input, can now be used to determine the dimensions of the zone. It is required that the user define the ratio of the total zone surface area to $A_{mod,zone}$. Equation (5.3.5) demonstrates its use.

$$A_{zone} = \gamma A_{mod,zone} \tag{5.3.5}$$

A_{zone} is the surface area of a zone and γ is the user defined ratio for zone area to modules in a zone area. γ is always greater than or equal to one by its definition. If it is defined as one, then there is no insulation surrounding the modules within a zone. If it is greater than one, there is considered to be insulation surrounding the modules and the insulation is adjacent to the modules on all sides. It fills in the lengths and widths not defined as module length or width. The exact length and width of each stretch of insulation is not needed to be known. This is because the area of the combined insulation is used to find the insulation thermal resistance, R_{ins}, for each zone. Area of the insulation, A_{ins}, is found using equation (5.3.6).

$$A_{ins} = A_{zone} - A_{mod,zone} \tag{5.3.6}$$

Two more important calculations that define zone dimensions are needed. They are length and width of the zone itself, insulation included. Now that area of a zone is known and the total

module length to total module width ratio is known, it is rather simple to calculate zone length and zone width. Equations (5.3.7) and (5.3.8) demonstrate this.

$$L_z = \sqrt{\beta_{lw} A_{zone}} \tag{5.3.7}$$

$$w_z = \sqrt{(1/\beta_{lw}) A_{zone}} \tag{5.3.8}$$

L_z is the length of an entire zone and w_z is the width of a zone. These are used throughout the calculations of the finned heat exchangers and are necessary for defining various fin dimensions. Their use can be seen throughout the next couple sections.

5.3.1 Rectangular Straight Fins

Rectangular straight fins are very common fin geometry because of their simplicity to manufacturer. They are commonly available in different sizes and can be mounted to other structures fairly easily. Their simplicity also makes for a good starting place in describing their relation to the methods used in TEPSS. The initial set up of the fins and their geometry requires a few input parameters. These parameters are used throughout the development of the rectangular straight fins and include the number of fins (N_f), the thickness of an individual fin (t_f), the length an individual fin protrudes from its base (L_f), and the thickness of the base (t_b). The only material property that is needed with regards to the fins is the conductive coefficient (k_{fin}).

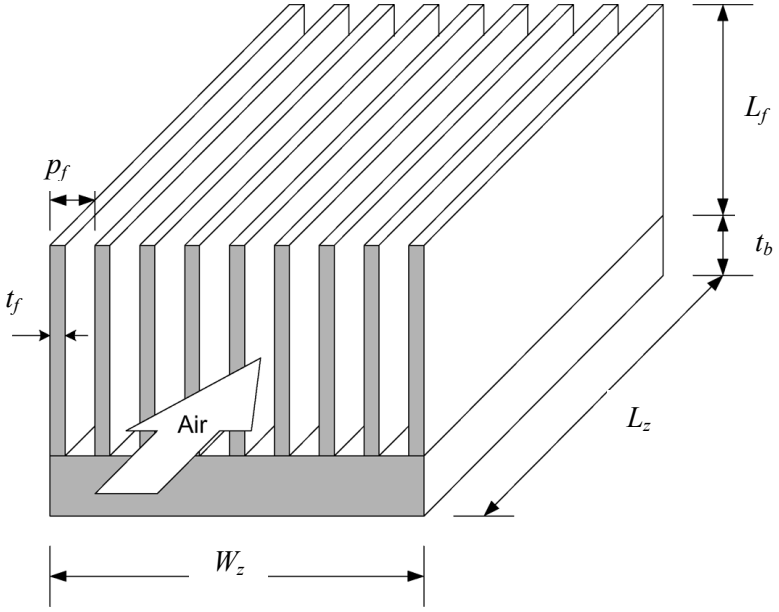


Figure 5.8: Visual of rectangular straight fins [48].

Equally important as the number of fins is the number of channels, N_{ch}. This is because the number of channels provides the area that allows fluid flow to pass through the heat exchanger. The number of channels is simply equal to the number fins minus one which can be seen in

equation (5.3.9). This is because two fins are considered the outer wall of the shell of the heat exchanger.

$$N_{ch} = N_f - 1 \tag{5.3.9}$$

The pitch of a fin is needed to be known to help determine the spacing between fins. The pitch helps to keep the fins constrained to the size of the heat exchanger. This is done by subtracting the thickness of one fin from the width of a zone, w_z. The obtained value is then divided by the number of channels which represents the pitch of one fin, p_f as seen in equation (5.3.10).

$$p_f = \frac{w_z - t_f}{N_{ch}} \tag{5.3.10}$$

The pitch is then used to find the spacing between fins, S_f. The spacing between fins is used for several calculations because it represents part of the dimensioning of the flow path. It is found in equation (5.3.11) by subtracting the thickness of a fin from the pitch of a fin.

$$S_f = p_f - t_f \tag{5.3.11}$$

Now that the spacing of the fin is known, it is possible to find the wetted perimeter, P_{wet}, of a flow path or one channel created by the fins. The wetted perimeter is found by adding two lengths of the fin plus two lengths of the spacing of a fin which can be seen in equation (5.3.12).

$$P_{wet} = 2L_f + 2S_f \tag{5.3.12}$$

Having calculated the wetted perimeter, it is now possible to find the hydraulic diameter, D_h. The hydraulic diameter is needed to provide the ability to use certain calculations that are typically dependent on a diameter. It is an artificial diameter representing the channel in which flow travels through. Equation (5.3.13) displays this calculation.

$$D_h = \frac{4L_f S_f}{P_{wet}} \tag{5.3.13}$$

An entrance area, A_{ent}, needs to be considered for the flow paths through the rectangular straight fins. This is equal to the spacing between the fins times the length of a fin which represents the size of the channel. The size of the channel is then multiplied by the number of channels to account for the entire flow through the heat exchanger which can be seen in equation (5.3.14).

$$A_{ent} = S_f L_f N_{ch} \tag{5.3.14}$$

The characteristic length of the fin, $L_{f,char}$, is represented by equation (5.3.15). It is equal to the length of the fin plus half of a fin thickness. This is needed for fin efficiency calculations.

$$L_{f,char} = L_f + \frac{t_f}{2} \tag{5.3.15}$$

The perimeter of the face of the fin, P_{face}, or the perimeter of the cross section of the fin is equal to two times the thickness of a fin plus two times the length of a zone. This parameter can be seen in equation (5.3.16) and is needed to assist in finding the efficiency of the proposed fin.

$$P_{face} = 2t_f + 2L_z \tag{5.3.16}$$

The cross sectional area of the fin, A_c, is the thickness of the fin times the length of a zone. This value is also used in calculating the efficiency of the designed fin and displayed in equation (5.3.17).

$$A_c = t_f L_z \tag{5.3.17}$$

The total surface area of all the fins, $A_{f,surf}$, is needed for finding the total surface area that is affected by convection and is demonstrated in equation (5.3.18). The combined fin surface area is equal to two times the number of channels to represent each of the fin surfaces within the flow path. This is multiplied by the characteristic length of the fins and then multiplied by the length of a zone to represent the entire fin surface area.

$$A_{f,surf} = 2N_{ch}L_{f,char}L_z \tag{5.3.18}$$

The total area of the base, $A_{b,surf}$, is also needed to help find the total effective surface area. The base surface area is simply the surface area of a zone minus the cross sectional area of each fin as seen in equation (5.3.19).

$$A_{b,surf} = A_{zone} - A_c N_f \tag{5.3.19}$$

Total effective surface area, $A_{tot,surf}$, is the area which fluid flow occurs and convective heat transfer is present. This is simply the surface areas of the fins and the base added together represented in equation (5.3.20).

$$A_{tot,surf} = A_{f,surf} + A_{b,surf} \tag{5.3.20}$$

Now that the fins have been designed, their performance needs to be evaluated. The first step to evaluating the fins is identifying how to find the Reynolds number using the parameters that are known at this point. The definition of Reynolds number is the ratio of inertial forces to viscous forces which is represented by multiplying the density (ρ) by the mean velocity (V) by the hydraulic diameter (D_h) and then dividing by viscosity (μ).

$$Re = \frac{\rho V D_h}{\mu} \tag{5.3.21}$$

Since the mean velocity has not yet been determined, it must be substituted. Mass flow rate divided by the entrance area is equal to density times the mean velocity. The occurrence of the substitution can be seen in (5.3.22).

$$Re = \frac{\dot{m} D_h}{A_{ent} \mu} \tag{5.3.22}$$

Because hydraulic diameter and entrance area are dependent on some of the same variables, they are substituted into equation (5.3.22) which can be seen in (5.3.23).

$$Re = \frac{\dot{m} \frac{4 L_f S_f}{P_{wet}}}{S_f L_f N_{ch} \mu} \tag{5.3.23}$$

Simplifying equation (5.3.23) yields equation (5.3.24) which is used to find the Reynolds number for this type of fins.

$$Re = \frac{4\dot{m}}{\mu P_{wet} N_{ch}} \tag{5.3.24}$$

Reynolds number is a necessary dimensionless parameter used in determining the performance of the proposed fin geometry. Prandtl number, *Pr*, is the ratio of the momentum and thermal diffusivities and another necessary dimensionless parameter needed for rectangular straight fin analysis. It is represented by equation (5.3.25) where k_{fluid} is the conductive coefficient of the fluid flowing through the heat exchanger. This dimensionless number has conceptual meaning in that at low Prandtl numbers, the conductive heat transfer is dominant and at high Prandtl numbers, the convective heat transfer is dominant.

$$Pr = \frac{C_p \mu}{k_{fluid}} \tag{5.3.25}$$

The first application of the Reynolds number is its use in finding the friction factor. When Reynolds is less than or equal to 3000, the flow is considered to be laminar. To determine the friction factor, *f*, data was taken from Kays and Crawford [41]. The aspect ratio, χ, of the channel dimensions is found making it so that it is always greater than one. This means that the length of the fin is compared to the spacing between the fins and the greater value is divided by the lesser value. A correlation for the tabulated data was found and seen in equation (5.3.26).

$$fRe = -0.4673\chi^2 + 7.8663\chi + 49.006 \tag{5.3.26}$$

The developed correlation is only valid for aspect ratios greater than or equal to one and less than or equal to eight. For aspect ratios greater than eight, *fRe* is equal to 96 and then divided by the Reynolds number to determine the friction factor using equation (5.3.27). For aspect ratios in the appropriate range, equation (5.3.26) is solved and the resulting value is divided by the Reynolds number as seen in equation (5.3.27).

$$f = \frac{fRe}{Re} \tag{5.3.27}$$

With the solved for Reynolds and Prandtl numbers, Nusselt number can now be found. Nusselt number, *Nu*, is the ratio of convection to pure conduction heat transfer. It is necessary to solve

for *Nu* in order to find the convective coefficient. Equation (5.3.28) is from [42] and is valid for laminar flow, combined thermal and velocity entry, $0.6 \leq Pr \leq 5$, $0.0044 \leq \mu/\mu_s \leq 9.75$, and uniform surface temperature. The surface temperature being considered constant is accurate for this analysis as each zone is considered isothermal and the changes in temperature are step changes by zone. The surface viscosity, μ_s, is needed for determining *Nu* with this equation.

$$Nu = 1.86\left(\frac{RePr}{L_z/D_h}\right)^{1/3}\left(\frac{\mu}{\mu_s}\right)^{0.14} \tag{5.3.28}$$

A different set of equations are needed if it is determined that the flow is turbulent. For turbulent, fully developed flow with a Reynolds number greater than or equal to 3000 and less than or equal to $5x10^6$, equation (5.3.29) is used to find the friction factor [43]. This is typically used for geometries with smooth surfaces which is the assumption that must be made with regards to the paths channeling the flow of the fluid across the fins. For non-smooth surfaces, the Moody diagram is typically used but cannot be transferred to the MATLAB® computer code so the assumption of smoothness needed to be made which is fairly accurate.

$$f = (0.790\ln Re - 1.64)^{-2} \tag{5.3.29}$$

Equation (5.3.30) [44] is used to find Nusselt number for turbulent flow through rectangular straight fins. This equation is typically used for circular tubes but the use of the hydraulic diameter has extended its use to the rectangular geometry. It is reasonable to use this correlation with a Prandtl number greater than 0.7. The condition of use for flow in circular tubes has $0.5 \leq Pr \leq 2000$. Other conditions include fully developed and $3000 \leq Re \leq 5x10^6$. The fully developed condition may add slight error to the analysis because it is assumed and not actually fully developed flow. It is required to keep the analysis practical though. For Reynolds numbers greater than the limit of this equation, another correlation is needed or the analysis cannot be completed.

$$Nu = \frac{(f/8)(Re-1000)Pr}{1+12.7(f/8)^{1/2}\left(Pr^{2/3}-1\right)} \tag{5.3.30}$$

Having solved the Nusselt number for either laminar or turbulent flow depending on the conditions, it is now possible to determine the convective coefficient, *h*. The equation that relates the two parameters can be seen in equation (5.3.31).

$$h = \frac{Nuk_{fluid}}{D_h} \tag{5.3.31}$$

In order to find the fin efficiency, equations (5.3.32) and (5.3.33) are needed. They come from Incropera et al. [37] and are suitable for fins of uniform cross sectional area. *m* comes from a coefficient in a second order differential equation pertaining to an energy balance of conduction and convection in a extended surface. η_f is the efficiency of one fin with the previously calculated fin parameters and provided inputs.

$$m = \sqrt{\frac{hP_{face}}{k_{fin}A_c}} \tag{5.3.32}$$

$$\eta_f = \frac{\tanh(mL_{f,char})}{mL_{f,char}} \tag{5.3.33}$$

With the efficiency of one fin, the efficiency of an array of fins can now be found. The overall surface efficiency, η_0, is the efficiency of the array of fins as well as the base surface to which the fins are attached [37]. This requires the use of the previously calculated surface area of all of the fins and the calculated surface area of the base plus the array of fins. The overall surface efficiency is calculated using equation (5.3.34).

$$\eta_0 = 1 - \left(\frac{A_{f,surf}}{A_{tot,surf}}\right)(1 - \eta_f) \tag{5.3.34}$$

The overall surface efficiency can be used towards finding the overall thermal resistance of the fins but first the thermal conductive resistance of the base, R_b, is needed. It is assumed that the base is made of the same material as the fins so k_{fin} is used. The area that heat is conducted through is the same area as the zone. This can be seen in equation (5.3.35).

$$R_b = \frac{t_b}{k_{fin}A_{zone}} \tag{5.3.35}$$

One more resistance is needed before the overall thermal resistance of the fins can be found. This resistance is the spreading resistance, R_{sp}, which was described in section 4.2. An effective area, A_{eff}, is needed for finding an effective convective coefficient, h_{eff}. The effective area is equal to the overall surface efficiency provided by equation (5.3.34) times the total surface area provided by equation (5.3.20) which can be seen in equation (5.3.36).

$$A_{eff} = \eta_0 A_{tot,surf} \tag{5.3.36}$$

The effective area can now be used to find the effective convective coefficient. It requires that the convective coefficient found in equation (5.3.31) be multiplied by the effective area divided by the area of a zone. This is displayed in equation (5.3.37).

$$h_{eff} = h\frac{A_{eff}}{A_{zone}} \tag{5.3.37}$$

R_{sp} requires inputs of effective convective coefficient (h_{eff}), fins base thickness (t_b), fin conductive coefficient (k_{fin}), and dimensions of the affected area by each individual module [14]. The spreading resistance is calculated for one module and then divided by the number of modules in a zone to capture the entire spreading resistance in a zone. The modules are thermally in parallel so an individual spreading resistance is equated as parallel resistances would typically be.

One of the two final values that are necessary to complete the analysis for rectangular straight fins is the thermal resistance of the fin array, R_f. This includes the convection and conduction of the fins, spreading resistance created by the TEG's interface, and conduction through the base. Equation (5.3.38) captures this necessary resistance for the system equations.

$$R_f = \frac{1}{\eta_0 h A_{tot,surf}} + R_{sp} + R_b \tag{5.3.38}$$

Before the reciprocal of the value obtained in (5.3.38) can be computed into a UA-value, thermal contact resistance needs to be considered. The thermal contact resistance, $R''_{th,con}$, is a user provided value and given in units that are equivalent to thermal resistance times unit area. This given variable represents any thermal paste or other substance used to reduce contact resistance of the mated fin bases and thermoelectric modules. This means that the thermal contact resistance should be divided by the total area of the modules within the zone, $A_{mod,zone}$. This thermal resistance is in series with R_f. They are added together and their reciprocal is taken to provide the UA-value for that side of the heat exchanger. Equation (5.3.39) shows this calculation. The obtained value for each side of the heat exchanger that utilizes this type of fins is used in the system equations from section 5.1.

$$UA = \frac{1}{R_f + \frac{R''_{th,con}}{A_{mod,zone}}} \tag{5.3.39}$$

The pressure drop caused by the designed fins is still required to complete the analysis. More surface area generally means more heat transfer but also means more pressure drop. There is a trade-off between thermal effects and flow effects. Pressure drop is dependent on the velocity of the fluid flow. Velocity can simply be calculated using equation (5.3.40). It captures the flow through all of the channels because A_{ent} is used.

$$V = \frac{\dot{m}}{\rho A_{ent}} \tag{5.3.40}$$

With the calculated velocity, change in pressure, ΔP, can be calculated. The greater the length of a zone, the greater the pressure drop will be. This is true of velocity as well but velocity is squared in equation (5.3.41) so its affect is even greater. Also, greater friction factors and densities can have negative effects. However, the larger hydraulic diameter can have a positive effect on the change in pressure, or pressure drop, across the fins.

$$\Delta P = f \frac{\rho V^2}{2 D_h} L_z \tag{5.3.41}$$

A similar method is used to find the UA-value and change in pressure for other fin geometries. Offset strip fins require the geometry be fully developed and parameters accounted for. Following the geometric calculations, performance calculations are completed to the overall fin thermal resistance and pressure drop.

5.3.2 Offset Strip Fins

Offset strip fins are common heat sink geometry because of their heat transfer capabilities. Each row of rectangular fins is offset from the previous row to create a staggered pattern for the flow path. This typically creates a greater pressure drop but also allows for a substantially larger heat transfer coefficient. The initial set up of the fins and their geometry requires a few input parameters. These parameters are used throughout the development of the offset strip fins and include the thickness of an individual fin (t_f), the length an individual fin protrudes from its base (L_f), and the thickness of the base (t_b) like the rectangular straight fins. The only material property that is needed with regards to the fins is the conductive coefficient (k_{fin}) like the rectangular straight fins. There are two additional parameters that are needed which differ from the previously described fin type. These parameters include the number of rows of fins (N_{rows}) and the number of fins in a row across the heat exchanger surface in the transverse direction ($N_{f,trans}$).

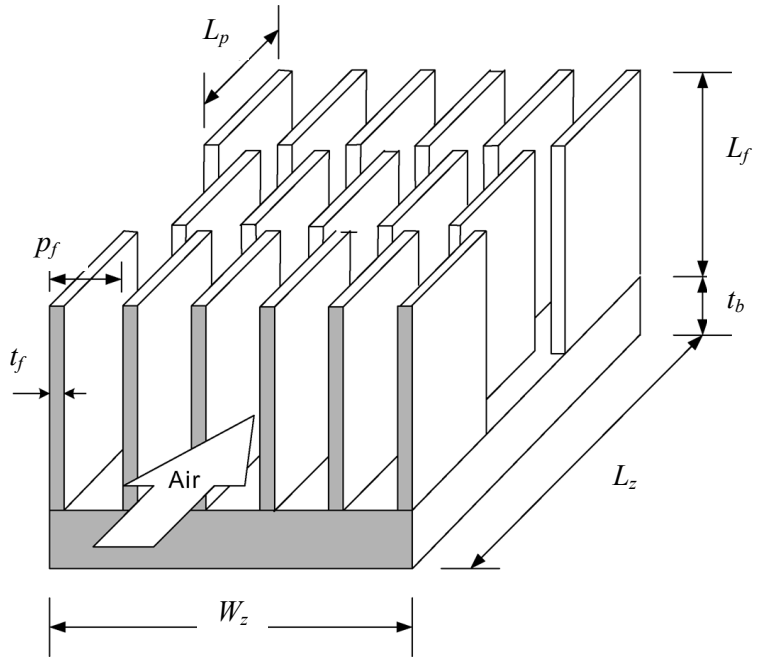


Figure 5.9: Visual of offset strip fins [48].

Equally important as the number of fins in a row is the number of channels representing entrance paths, $N_{ch,ent}$. This is because the number of channels provides the area that allows fluid flow to pass through the heat exchanger. The initial channels only extend the length of the first section of the fins so the assumption exists that the entrance air is evenly distributed through the remainder of the offset strip fins. The number of channels is simply equal to the number fins in the transverse direction minus one which can be seen in equation (5.3.42). This is because the outer two fins in the first row are considered to be flush to the heat exchanger shell wall. Starting with the first row, every odd numbered row's outer fins are flush to the heat exchanger wall. The second row and every even numbered row are indented from the wall and have one less fin than

the first row. The shell is not considered to affect performance and exists as a way of keeping the heat exchanger fluid inside the heat exchanger.

$$N_{ch,ent} = N_{f,trans} - 1 \tag{5.3.42}$$

The pitch of a fin is needed to be known to help determine the spacing between fins. The pitch helps to keep the fins constrained to the size of the heat exchanger. This is done by subtracting the thickness of one fin from the width of a zone, w_z. The obtained value is then divided by the number of channels at the flow entrance which represents the pitch of one fin, p_f as seen in equation (5.3.43). This method for finding the pitch is the same as for finding the pitch of the rectangular straight fins.

$$p_f = \frac{w_z - t_f}{N_{ch,ent}} \tag{5.3.43}$$

The pitch is then used to find the spacing between fins, S_f. It is found in equation (5.3.11) by subtracting the thickness of a fin from the pitch of a fin as it was for the rectangular straight fins.

Now that the pitch is known, the contraction ratio, ε, can be determined by equation (5.3.44). This ratio is used for evaluating pressure drop but is calculated here because it is representative of a physical property based on dimensions.

$$\varepsilon = \frac{p_f - t_f}{p_f} \tag{5.3.44}$$

The offset strip length, L_p, can now be calculated by using equation (5.3.45). It is the length of one strip of fin and is found by dividing the zone length by the number of rows. This ensures that each strip fin is the same length so the performance of each fin is equal.

$$L_p = \frac{L_z}{N_{rows}} \tag{5.3.45}$$

The hydraulic diameter, D_h, requires a complicated equation [45], seen in (5.3.46), to represent the complexity of the offset strip fin. It factors in all of the dimensions related to the fin including S_f, L_f, L_p, and t_f. Offset strip fin hydraulic diameter is found with the same basis as typical hydraulic diameters are found but in this case, the channel surface areas are considered. Also, the fin edges, both vertical and lateral, are included. This equation essentially becomes a volume encompassed by two strip fins divided by the surrounding surface areas.

$$D_h = \frac{4S_f L_f L_p}{2(S_f L_p + L_f L_p + t_f L_f) + S_f t_f} \tag{5.3.46}$$

An entrance area, A_{ent}, needs to be considered for the flow paths through the offset strip fins. This is equal to the spacing between the fins times the length of a fin which represents the size of the channel. The size of the channel is then multiplied by the number of channels in the region of

the entrance row to account for the entire flow through the heat exchanger which can be seen in equation (5.3.47). This equation is very similar to equation (5.3.14) for rectangular straight fins.

$$A_{ent} = S_f L_f N_{ch,ent} \tag{5.3.47}$$

The characteristic length of the fin, $L_{f,char}$, is represented by equation (5.3.15). It is equal to the length of the fin plus half of a fin thickness. This is needed for fin efficiency calculations and is the same as the characteristic length for rectangular straight fins.

The perimeter of the face of the fin, P_{face}, or the perimeter of the cross section of the fin is equal to two times the thickness of a fin plus two times the length of an individual offset strip fin. This parameter can be seen in equation (5.3.48) and is needed to assist in finding the efficiency of the proposed fin.

$$P_{face} = 2t_f + 2L_p \tag{5.3.48}$$

The cross sectional area of the fin, A_c, is the thickness of the fin times the length of an individual strip fin. This value is also used in calculating the efficiency of the designed fin and displayed in equation (5.3.49).

$$A_c = t_f L_p \tag{5.3.49}$$

The total surface area of all the fins, $A_{f,surf}$, is needed for finding the total surface area that is affected by convection. Before this can be done, the number of fins, N_f, needs to be calculated. This requires the number of rows to be multiplied by the number of transverse fins. However, the result doesn't account for there being one less fin in every even numbered row. Equation (5.3.50) shows the appropriate fins to be subtracted the previous multiplication which is dependent on whether the number of rows is odd or even.

$$N_f = \begin{Bmatrix} N_{rows}N_{f,trans} - \frac{N_{rows}-1}{2} & \middle| & N_{rows}\ is\ an\ odd\ integer \\ N_{rows}N_{f,trans} - \frac{N_{rows}}{2} & \middle| & N_{rows}\ is\ an\ even\ integer \end{Bmatrix} \tag{5.3.50}$$

Now, the total surface area of all the fins can be calculated using equation (5.3.51). The combined fin surface area is equal to two times the number of channels to represent each of the fin surfaces within the flow path. This is multiplied by the characteristic length of the fins and then multiplied by the length of an offset strip to represent the entire fin surface area.

$$A_{f,surf} = 2N_f L_{f,char} L_p \tag{5.3.51}$$

The total area of the base, $A_{b,surf}$, is also needed to help find the total effective surface area. The base surface area is simply the surface area of a zone minus the cross sectional area of each fin as seen in equation (5.3.19) from the rectangular straight fin analysis. Total effective surface area, $A_{tot,surf}$, is the area which fluid flow occurs and convective heat transfer is present. This is simply

the surface areas of the fins and the base added together similar to the rectangular straight fins represented in equation (5.3.20) from subsection 5.3.1.

Now that the fins have been designed, their performance needs to be evaluated. Mass flow rate is considered a known parameter as it is input into the system analysis. Mass flow rate and equation (5.3.40) can be used to determine velocity, *V*. Velocity of the fluid flow is needed to find the Reynolds number, *Re*. The common form of the Reynolds number, equation (5.3.21) is used for this purpose. Prandtl number, *Pr*, is also needed to complete the analysis. Equation (5.3.25) can be used to find it. Equations from the analysis by Manglik and Bergles [45] are dependent on Reynolds number and Prandtl number.

Manglik and Bergles [45] took experimental data from Joshi and Webb [46] and Kays and London [47] with regards to offset strip fins. A multivariable regression analysis was performed and yielded laminar and turbulent flow correlation coefficients with 0.99 confidence level for the Colburn factor, *j*, and friction factor, *f*, correlations. The regression fits for the experimental data do not exhibit any characteristic discontinuity with respect to *Re* so they are used to develop a single equation to represent the laminar, transition, and turbulent flow regions. Four equations were developed to describe the laminar and turbulent region asymptotes for *f* and *j*. The laminar asymptotes describe $Re \rightarrow 0$ and the turbulent asymptotes describe $Re \rightarrow \infty$. Equations (5.3.52) and (5.3.56) were obtained for *j* and *f*, respectively. These equations correlate to the examined experimental data within ±20%. They describe the right trend in the heat transfer and friction loss behavior of offset strip fins in each of the flow regimes. This means that the flow regime does not need to be determined which ensures the use of these equations in design changes to the offset strip fins. These equations are applicable for fluids with Prandtl numbers in between 0.5 and 15.

$$j = 0.6522Re^{-0.5403}\left(\frac{S_f}{L_f}\right)^{-0.1541}\left(\frac{t_f}{L_p}\right)^{0.1499}\left(\frac{t_f}{S_f}\right)^{-0.0678} \times\left[1+5.269\times10^{-5}Re^{1.340}\left(\frac{S_f}{L_f}\right)^{0.504}\left(\frac{t_f}{L_p}\right)^{0.456}\left(\frac{t_f}{S_f}\right)^{-1.055}\right]^{0.1} \quad (5.3.52)$$

The Colburn j-factor obtained in equation (5.3.52) is based on log mean temperature difference (LMTD). In order to convert this to a useful value, equation (5.3.53) is introduced. This represents the Colburn j-factor relationship with the convective coefficient based in LMTD, h_{LMTD}.

$$h_{LMTD} = j(\rho V C_p)Pr^{-2/3} \quad (5.3.53)$$

The number of transfer units, *NTU*, is a dimensionless parameter that helps relate h_{LMTD} to the convective coefficient, *h*.

$$NTU = \frac{h_{LMTD} A_{tot,surf}}{\dot{m} C_p} \tag{5.3.54}$$

Equation (5.3.55) completes the effectiveness-NTU analysis of the offset fin strips and provides a result for the convective coefficient to be used in the remaining analysis [48].

$$h = \frac{\dot{m} C_p}{A_{tot,surf}} (1 - e^{-NTU}) \tag{5.3.55}$$

The average value for the convective coefficient is now ready for use with some familiar analysis performed for rectangular straight fins. The convective coefficient, h, can be plugged into equation (5.3.32) which is used to find the fin efficiency in equation (5.3.33).With the efficiency of one fin, the efficiency of an array of fins can now be found. The overall surface efficiency, η_0, can be found using equation (5.3.34). This can be used to find the effective area in equation (5.3.36) which is used to find the effective convective coefficient in equation (5.3.37) which, in turn, is used to find the spreading resistance. The thermal conductive resistance of the base of the fins is also needed and found using equation (5.3.35). There is now enough information to find the thermal resistance of the fin array, R_f, by utilizing equation (5.3.38). This variable can now be used for its ultimate purpose of find the value of UA in equation (5.3.39) for the specific side of the heat exchanger.

The pressure drop caused by the offset strip fins is still required to complete the analysis of the performance of the proposed fins. An equation for friction factor [45] was described above and can be seen in equation (5.3.56).

$$f = 9.6243 Re^{-0.7422} \left(\frac{S_f}{L_f}\right)^{-0.1856} \left(\frac{t_f}{L_p}\right)^{0.3053} \left(\frac{t_f}{S_f}\right)^{-0.2659}$$
$$\times \left[1 + 7.669 \times 10^{-8} Re^{4.429} \left(\frac{S_f}{L_f}\right)^{0.920} \left(\frac{t_f}{L_p}\right)^{3.767} \left(\frac{t_f}{S_f}\right)^{0.236}\right]^{0.1} \tag{5.3.56}$$

Inlet and exit loss coefficients, K_c and K_e, respectively, are needed to completely describe the change in pressure, ΔP, through the offset strip fins. Equations (5.3.57) and (5.3.58) were developed by Kim and Webb [48] by taking a curve fit of data from Kays and London [47] for $Re=\infty$. $Re=\infty$ is used for the offset strip fins because it is assumed that no boundary layer develops. This is true for smaller length of offset fins, L_p, but it is hard to say the exact range of operation so it is assumed for all lengths of offset fins.

$$K_c = [0.4 + 0.03775\varepsilon - 0.4362\varepsilon^2]/N_z \tag{5.3.57}$$

$$K_e = [(1 - \varepsilon)^2]/N_z \tag{5.3.58}$$

These equations utilize the contraction ratio, ε, determined in equation (5.3.44). The curve fits are divided by the number of zones, N_z, because this will evenly distribute the inlet and exit

losses to each zone. The total pressure losses are found by multiplying the pressure losses by the number of zones. This will then only count the inlet and exit losses once total.

With all of the friction and loss coefficients calculated, change in pressure can be calculated. Velocity is squared in equation (5.3.59) and affects each loss coefficient so its affect on pressure drop is greater than any other factor.

$$\Delta P = \left(K_c + K_e + \frac{4fL_z}{D_h}\right)\frac{\rho V^2}{2} \tag{5.3.59}$$

This method was similar to the method used to find the UA-value and change in pressure for other rectangular straight fins. Following the geometric calculations, performance calculations are completed to the overall fin thermal resistance and pressure drop for offset strip fins. The same procedure is used for aligned and staggered pin fins as described in the next section.

5.3.3 Aligned and Staggered Pin Fins

Pin fins are circular extrusions protruding from the base of their associated heat exchanger. Small pin fin heat sinks applications include cooling of electronic or computer components. Pin fins resemble larger scale tube banks that are typically used in industrial applications. Banks of tubes generally have one fluid move over the tubes, while another fluid passes through the tubes to exchange heat [37]. Flow across the tube banks is analyzed as an external flow and several equations and data tables are provided [37]. Internal fluid flow within the tube is ignored in this analysis because the tubes are treated as pin fins which are considered to be a solid structure.

Pin fins are either aligned as seen in Figure 5.10 or staggered as seen in Figure 5.11. Each of these configurations requires different equations to analyze as their performance is not the same. Convective heat transfer and flow properties are affected by boundary layer separation effects and wake interactions. Heat transfer coefficients of pins in the first row are approximately equal to a single tube in cross flow. The first few rows of pin fins generate turbulence which increases heat transfer for pin fins downstream. Generally, by the fourth or fifth row, the convective coefficient changes only slightly from row to row. Analysis is performed to find the average convective coefficient for a pin fin arrangement.

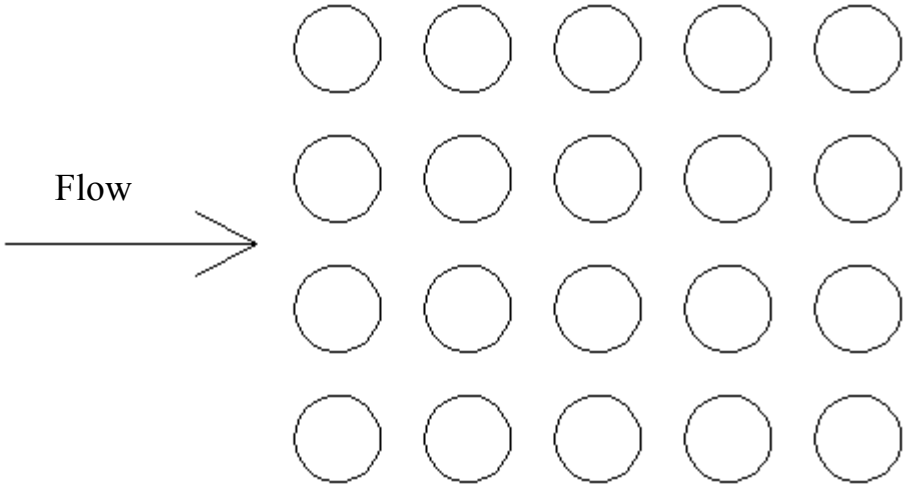


Figure 5.10: Example of aligned pin fins.

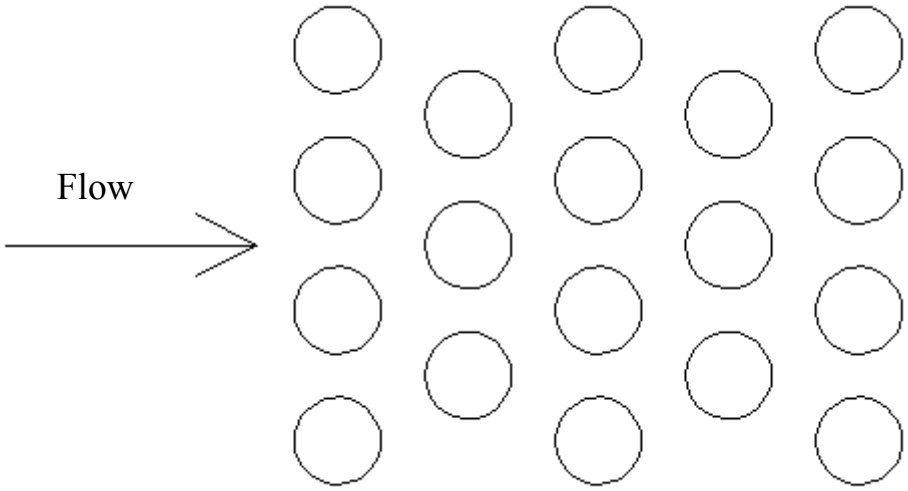


Figure 5.11: Example of staggered pin fins.

The initial inputs needed for a set of pin fins include whether the fins are aligned or staggered, the diameter of an individual fin (D_f), the length of a fin (L_f), the transverse pitch for the fins in the array (p_{trans}), and the longitudinal pitch of fins in the array (p_{long}). Also, the thickness of the base, t_b, is needed as it was in each of the other fin type configurations. And, the conductive coefficient, k_{fin}, is needed like in the two previous analyses.

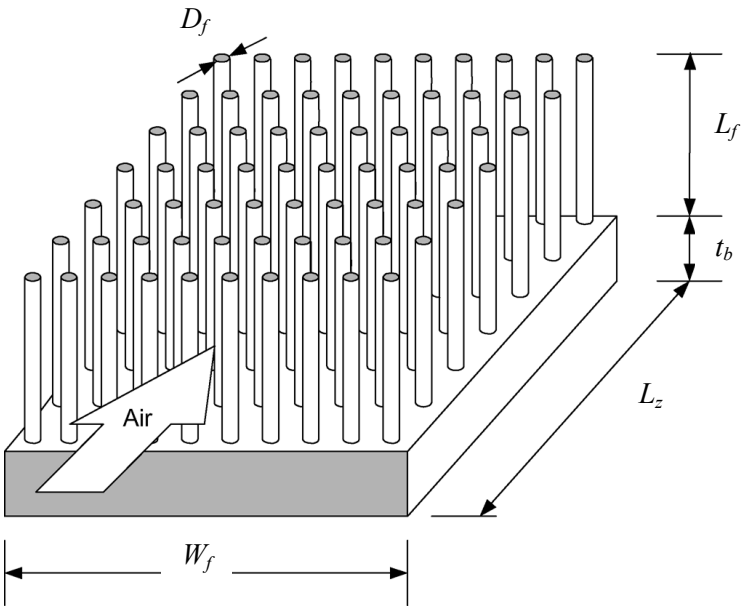


Figure 5.12: Visual of a pin fin array [48].

The characteristic length of the fin, $L_{f,char}$, is represented by equation (5.3.60). It is equal to the length of the fin plus a quarter of a fin diameter. This is needed for fin efficiency calculations.

$$L_{f,char} = L_f + \frac{D_f}{4} \tag{5.3.60}$$

The spacing between fins in transverse plane, $A1$, is calculated by taking the transverse pitch and subtracting the diameter of a fin. This is true for both the aligned and staggered configurations and can be seen in equation (5.3.61).

$$A1 = p_{trans} - D_f \tag{5.3.61}$$

The cross sectional area of a pin fin, A_c, is simply the area of a circle. This is demonstrated in equation (5.3.62).

$$A_c = \frac{\pi D_f^2}{4} \tag{5.3.62}$$

The wetted perimeter, P_{wet}, is set equal to the circumference of a circle as seen in equation (5.3.63). It is assumed that the entire circumference of the pin is wetted because it is hard to determine the exact amount of the surface exposed to flow. Typically, flow separation occurs around a cylinder but the point of separation can occur in a wide range of locations depending on flow conditions. Also, depending on the configuration, swirling vortices from downstream pins may act on upstream pins. The unpredictability of the situation requires the wetted perimeter to be overestimated and some inaccuracy can be associated with this assumption.

$$P_{wet} = \pi D_f \tag{5.3.63}$$

A calculation for the number of rows, N_{rows}, was needed on the basis that there would be only whole fins and no partial or cut off fin pieces. When the calculation performed in equation (5.3.64) is rounded down to the nearest whole number, a value is provided for the number of rows that meets the whole fin criteria. This will potentially provide some underutilized flow space at the end of the heat exchanger but it provides more accurate results as a realistic and correct number of rows is used for the remaining calculations.

$$N_{rows} = \frac{L_z - D_f}{p_{long}} + 1 \tag{5.3.64}$$

The number of columns, $N_{columns}$, of pin fins is calculated in the same manner as the number of rows was in equation (5.3.65). It is dependent on the width of the zone and the transverse pitch as it is in the direction perpendicular to the previous calculation for the number of rows. It should be noted the first row always has a greater number of pin fins than the second row if the configuration is staggered. The number of columns is based off of the row with the greater number of fins or any odd numbered row.

$$N_{columns} = \frac{w_z - D_f}{p_{trans}} + 1 \tag{5.3.65}$$

With the number of columns for the proposed design known, an entrance area, A_{ent}, needs to be considered for the flow paths past the first row of pin fins. This is equal to the entire entrance area dimensions minus the front cross section of each of the fins which is represented by equation (5.3.66). The flow is channeled around the front face of the pin fins and considered the entrance area into the heat exchanger.

$$A_{ent} = w_z L_f - N_{columns} D_f L_f \tag{5.3.66}$$

The number of columns and number of rows are now known parameters so the total number of fins, N_f, can be found. If the pin fins are aligned, the number of fins in the array can be calculated simply by multiplying the number of columns by the number of rows as seen in equation (5.3.67). If the pins are staggered, then finding the number of fins requires a slightly more complicated approach. Equation (5.3.68) demonstrates the necessary approach to finding the total number of fins whether the number of rows is odd or even. It is required to subtract a value from the number of rows times the number of columns because every even numbered row has one less fin than the odd numbered rows.

$$N_f = N_{columns} N_{rows} \tag{5.3.67}$$

$$N_f = \begin{Bmatrix} N_{columns} N_{rows} - \frac{N_{rows}-1}{2} \\ N_{columns} N_{rows} - \frac{N_{rows}}{2} \end{Bmatrix} \begin{Bmatrix} N_{rows}\ is\ an\ odd\ integer \\ N_{rows}\ is\ an\ even\ integer \end{Bmatrix} \tag{5.3.68}$$

A staggered array of pin fins requires a couple additional equations to determine spacing properties and selection of the proper equations for use. The diagonal pitch, p_d, is the relationship between the center of the pins in one row to the center of the pins in the next row. It forms a right angle triangle with the other pitches and the Pythagorean Theorem can be used to determine its length as seen in equation (5.3.69).

$$p_d = \sqrt{p_{long}^2 + \left(\frac{p_{trans}}{2}\right)^2} \tag{5.3.69}$$

The diagonal flow path, *A2*, is along the same axis as the diagonal pitch, p_d. The flow path, *A2*, is simply the diagonal pitch minus two half diameters of the pins which is equal to a whole diameter being subtracted from the pitch. This is shown in equation (5.3.70).

$$A2 = p_d - D_f \tag{5.3.70}$$

The total surface area of all the fins, $A_{f,surf}$, is needed for finding the total surface area that is affected by convection and is demonstrated in equation (5.3.71). This equation holds true for both the aligned and staggered pin fin configurations. The combined fin surface area is equal to the circumference times the characteristic length times the total number of fins.

$$A_{f,surf} = \pi N_f L_{f,char} D_f \tag{5.3.71}$$

The total area of the base, $A_{b,surf}$, is also needed to help find the total effective surface area. The base surface area is simply the surface area of a zone minus the cross sectional area of each fin as seen in equation (5.3.19) from the rectangular straight fin analysis. Total effective surface area, $A_{tot,surf}$, is the area which fluid flow occurs and convective heat transfer is present. This is simply the surface areas of the fins and the base added together similar to the rectangular straight fins represented in equation (5.3.20) from subsection 5.3.1.

Now that the fins have been designed, their performance needs to be evaluated. Prandtl number, *Pr*, is needed to complete the analysis. Equation (5.3.25) can be used to find it. The Prandtl number based on surface level properties, Pr_s, is also needed for to equations developed by [49]. It is demonstrated in equation (5.3.72).

$$Pr_s = \frac{C_{p,s}\mu_s}{k_s} \tag{5.3.72}$$

$C_{p,s}$ is the specific heat capacity at the surface, μ_s is the viscosity at the surface, and k_s is the conductive coefficient at the surface. These properties are determined based on temperatures T_h or T_c from the system equations based on which side of the heat exchanger is being analyzed. There is some error in using these temperatures as they are the surface temperatures of the thermoelectric modules and not of the fins. These temperatures are used because they are available for iterations by the TEPSS program after the initial iteration is stepped through. Prandtl number, *Pr*, is based on the arithmetic mean of the inlet and outlet temperatures of the

zone so a value more closely related to the surface was needed. A direct surface temperature relationship is preferable but there is not an easy way to determine an exact value for this without complicating the TEPSS code. TEPSS is intended for use on regular computers and should not require a supercomputer to complete the calculations in a timely manner. This aspect of TEPSS is discussed more in Chapter 6.

Mass flow rate is considered a known parameter as it is input into the system analysis. Mass flow rate and equation (5.3.40) can be used to determine the velocity, V. Velocity of the fluid flow is needed to find the maximum velocity, V_{max}, which is used to eventually obtain the maximum Reynolds number, Re_{max}. If the pin fins are aligned, the maximum velocity can be found using equation (5.3.73).

$$V_{max} = \frac{p_{trans}}{A1} V \tag{5.3.73}$$

If the pin fins are staggered, equation (5.3.74) is applied. A comparison is performed relating p_d to p_{trans} and D_f. Equation (5.3.74) shows which equation is to be used to determine V_{max}. The maximum velocity is the same as equation (5.3.73) when it occurs at the transverse plane, $A1$. The maximum velocity is equal to the second part of equation (5.3.74) when it occurs at the diagonal plane, $A2$.

$$V_{max} = \begin{cases} \frac{p_{trans}}{A1} V \middle| p_d \geq (p_{trans} + D_f)/2 \\ \frac{p_{trans}}{2 \cdot A2} V \middle| p_d < (p_{trans} + D_f)/2 \end{cases} \tag{5.3.74}$$

For both the aligned and staggered arrays, the maximum velocity can be used to determine the maximum Reynolds number as seen in equation (5.3.75). The common form of the Reynolds number, equation (5.3.21) is used but with the appropriate values substituted to represent the appropriate analysis.

$$Re_{max} = \frac{\rho V_{max} D_f}{\mu} \tag{5.3.75}$$

Now that the maximum Reynolds number has been solved for, it is possible to determine which equation for average Nusselt number, $\overline{Nu}$, should be used. Equation (5.3.76) is used for maximum Reynolds numbers that fall in the range of $10^2 - 10^3$. The equation is actually recommended for all $RePr \geq 0.2$ and covers the entire range of available data for Reynolds number and a wide range of the available data for the Prandtl number [50]. It represents external flow across a single isolated cylinder and is recommended over the other equation for average Nusselt number for the range of Reynolds numbers mentioned above. Its use is intended for properties evaluated at the film temperature; however, there is uncertainty in the film temperature properties as they are based on the previous iteration's module temperature. It is best to accept some error by using Pr and the properties associated with the arithmetic mean of the inlet and outlet temperatures.

$$\overline{Nu} = 0.3 + \frac{0.62Re^{1/2}Pr^{1/3}}{\left[1+(0.4/Pr)^{2/3}\right]^{1/4}}\left[1+\left(\frac{Re}{282{,}000}\right)^{5/8}\right]^{4/5} \quad (5.3.76)$$

The second equation for average Nusselt number comes from Zukauskas [49] and is shown in equation (5.3.77). It is valid for $0.7 \leq Pr \leq 500$ and $1000 \leq Re_{max} \leq 2\ X\ 10^6$. All of the properties except Pr_s are evaluated at the arithmetic mean of the fluid inlet and outlet temperatures. Depending on the value of the maximum Reynolds number, constants C and b will change. The values for constants C and b are shown in Table 5.1 for aligned pin fins and Table 5.2 for staggered pin fins. Constants C and b are reported for Re_{max} for $10 - 10^2$ so they are accepted and used when necessary [37].

$$\overline{Nu} = CC_2Re_{max}^{b}Pr^{0.36}\left(\frac{Pr}{Pr_s}\right)^{1/4} \quad (5.3.77)$$

Table 5.1: Constants for equation (5.3.77) for aligned pin fins.

Re_{max}	C	b
$10 - 10^2$	0.80	0.40
$10^2 - 10^3$	Approximate as a single (isolated) cylinder	
$10^3 - 2\ X\ 10^5$ & $(p_{trans}/p_{long}) > 0.7$	0.27	0.63
$10^3 - 2\ X\ 10^5$ & $(p_{trans}/p_{long}) < 0.7$	Heat transfer is inefficient and aligned pin fins should not be used	
$2\ X\ 10^5 - 2\ X\ 10^6$	0.021	0.84

Table 5.2: Constants for equation (5.3.77) for staggered pin fins.

Re_{max}	C	b
$10 - 10^2$	0.90	0.40
$10^2 - 10^3$	Approximate as a single (isolated) cylinder	
$10^3 - 2\ X\ 10^5$ & $(p_{trans}/p_{long}) > 2$	0.40	0.60
$10^3 - 2\ X\ 10^5$ & $(p_{trans}/p_{long}) < 2$	$0.35(p_{trans}/p_{long})^{1/5}$	0.60
$2\ X\ 10^5 - 2\ X\ 10^6$	0.022	0.84

Equation (5.3.77) also has a correction factor, C_2, for number of rows of pin fins less than twenty. Because it is possible and highly likely that a zone has less than twenty rows, the correction factor is based on the total number of rows of pin fins in the heat exchanger, $N_{rows,tot}$. This is acceptable because the flow continues from the first zone to the second zone and so forth with minimal changes. This means that, if C_2 was determined by the number of rows in a zone, the performance of the heat exchanger would be degraded without reason. Therefore, equation (5.3.78) is introduced to determine the total number of rows in the heat exchanger. This value is then used in conjunction with Table 5.3 to determine the correction factor for aligned or

staggered pin fin arrays. A value for the correction factor provides enough information to determine the average Nusselt number and proceed with the analysis.

$$N_{rows,tot} = N_{rows}N_z \tag{5.3.78}$$

Table 5.3: Correction factor, C_2 for equation (5.3.77) for aligned and staggered pin fins.

$N_{rows,tot}$	*Aligned*	*Staggered*
1	0.70	0.64
2	0.80	0.76
3	0.86	0.84
4	0.90	0.89
5	0.92	0.92
6	0.935	0.935
7	0.95	0.95
8	0.9567	0.9567
9	0.9633	0.9633
10	0.97	0.97
11	0.9733	0.9733
12	0.9767	0.9767
13	0.98	0.98
14	0.9833	0.9833
15	0.9867	0.9867
16	0.99	0.99
17	0.9925	0.9925
18	0.9995	0.9995
19	0.9975	0.9975
20+	1.0	1.0

Having obtained the average Nusselt number, it is now possible to compute the average convective coefficient, $\bar{h}$ as displayed in equation (5.3.79).

$$\bar{h} = \frac{\overline{Nu}k_{fluid}}{D_f} \tag{5.3.79}$$

The average value for the convective coefficient is now ready for use with some familiar analysis performed for rectangular straight fins. The average convective coefficient can be plugged into equation (5.3.32) which is used to find the fin efficiency in equation (5.3.33).With the efficiency of one fin, the efficiency of an array of fins can now be found. The overall surface efficiency, η_0, can be found using equation (5.3.34). This can be used to find the effective area in equation (5.3.36) which is used to find the effective convective coefficient in equation (5.3.37) which, in

turn, is used to find the spreading resistance. The thermal conductive resistance of the base of the fins is also needed and found using equation (5.3.35). There is now enough information to find the thermal resistance of the fin array, R_f, by utilizing equation (5.3.38). This variable can now be used for its ultimate purpose of find the value of UA in equation (5.3.39) for the specific side of the heat exchanger. This part of the analysis requires the same procedure that was used for rectangular straight fins and offset strip fins. Any other proposed fin types should follow this routine to easily interact with the TEPSS computing process. This will be explained more in depth in Chapter 6.

Finding the change in pressure across the pin fin array is necessary to complete the analysis of this type of fins. The pressure drop is generally a function of the velocity and the friction factor. Friction factor is often determined from experimental data representing generic flow parameters. The experimental data is often correlated into an equation representing a range of acceptable parameters that it can be applied to. To determine the pressure drop across the pin fin array, a friction factor is required. However, friction factor in an equation form is not readily available for this type of finned heat exchanger. A different method needs to be applied which utilizes a reasonable substitute for the friction factor. The reasonable substitute that can be applied is the Hagen number [51, 54]. For both types of pin fins, the Hagen number is compiled of a laminar component and a turbulent component. For aligned pin fins, equation (5.3.80) shows this, and for staggered pin fins, equation (5.3.81) shows this.

$$Hg = Hg_{lam} + Hg_{turb}\left[1 - exp\left(1 - \frac{Re+1000}{2000}\right)\right] \tag{5.3.80}$$

$$Hg = Hg_{lam} + Hg_{turb}\left[1 - exp\left(1 - \frac{Re+200}{1000}\right)\right] \tag{5.3.81}$$

Hg represents the Hagen number, Hg_{lam} represents the laminar component of the Hagen number, and Hg_{turb} is the turbulent component of the Hagen number. The turbulent component is scaled based on the Reynolds number and has little to no effect on the flow when the Reynolds number is in the laminar regime. The turbulent component has a substantial effect as the Reynolds number approaches infinity and will dominate the equations based on the equation for Hg_{turb}.

In order to determine the Hagen number, several dimensionless variables are needed. The three pitches are divided by the diameter of the fin to provide three equations. Equation (5.3.82) represents the dimensionless transverse pitch, X_T, and equation (5.3.83) represents the dimensionless longitudinal pitch, X_L. Equation (5.3.84) represents the dimensionless diagonal pitch, X_d, which is used for the staggered pin fin arrays. Each of the dimensionless variables is substituted into equations for Hg_{lam} and Hg_{turb}.

$$X_T = \frac{p_{trans}}{D_f} \tag{5.3.82}$$

$$X_L = \frac{p_{long}}{D_f} \tag{5.3.83}$$

$$X_d = \frac{p_d}{D_f} \tag{5.3.84}$$

This method is valid for a specific range of operational parameters. The tube diameters from the experimental sources varied in the range of $7.9\ mm \leq D_f \leq 73\ mm$. For both, aligned and staggered pin fin arrays, the Hagen equations are valid for $1 \leq D_f \leq 300000$ and $N_{rows,tot} \geq 5$. For aligned pin fins, their development is confined to $1.25 \leq X_T \leq 3.0$ and $1.2 \leq X_L \leq 3.0$. For staggered pin fins, analysis is limited to $1.25 \leq X_T \leq 3.0$, $0.6 \leq X_L \leq 3.0$, and $X_d \geq 1.25$.

The laminar contributions to the Hagen number for aligned pin fin arrays can be seen in equation (5.3.85).

$$Hg_{lam} = 140Re\frac{\left(X_L^{0.5}-0.6\right)^2+0.75}{X_T^{1.6}\left(\frac{4X_TX_L}{\pi}-1\right)} \tag{5.3.85}$$

This equation is valid for staggered pin fins when V_{max} occurs in the transverse plane which can be seen in the first part of equation (5.3.86). The second part of (5.3.86) is the laminar component of the Hagen number when V_{max} occurs in the diagonal plane. It varies only in that $X_T^{1.6}$ is substituted with $X_d^{1.6}$.

$$Hg_{lam} = \left\{ \begin{matrix} 140Re\frac{\left(X_L^{0.5}-0.6\right)^2+0.75}{X_T^{1.6}\left(\frac{4X_TX_L}{\pi}-1\right)} \Big| when\ V_{max}\ occurs\ at\ A1 \\ 140Re\frac{\left(X_L^{0.5}-0.6\right)^2+0.75}{X_d^{1.6}\left(\frac{4X_TX_L}{\pi}-1\right)} \Big| when\ V_{max}\ occurs\ at\ A2 \end{matrix} \right\} \tag{5.3.86}$$

The turbulent component of the Hagen number, Hg_{turb}, for aligned pin fins can be seen in equation (5.3.87).

$$Hg_{turb} = \left\{ \left[0.11 + \frac{0.6\left(1-\frac{0.94}{X_L}\right)^{0.6}}{(X_T-0.85)^{1.3}} \right] \times 10^{0.47\left(\frac{X_L}{X_T}-1.5\right)} + 0.015(X_T-1)(X_L-1) \right\}$$

$$\times Re^{2-0.1\left(\frac{X_L}{X_T}\right)} + \phi_{t,n}Re^2 \tag{5.3.87}$$

$\varphi_{t,n}$ is the friction factor that accounts for the influence of inlet and outlet pressure losses. The method for calculating it can be seen in equation (5.3.88).

$$\phi_{t,n} = \left\{ \begin{matrix} \left[\frac{1}{2X_T^2}\left(\frac{1}{N_{rows,tot}}-\frac{1}{10}\right)\right]/N_z \\ \left[2\left[\frac{X_d-1}{X_T(X_T-1)}\right]^2\left(\frac{1}{N_{rows,tot}}-\frac{1}{10}\right)\right]/N_z \\ 0 \end{matrix} \middle| \begin{matrix} 5 \leq N_{rows,tot} \leq 10\ and\ when\ V_{max}\ occurs\ at\ A1 \\ 5 \leq N_{rows,tot} \leq 10\ and\ when\ V_{max}\ occurs\ at\ A2 \\ N_{rows,tot} > 10 \end{matrix} \right\} \tag{5.3.88}$$

$\varphi_{t,n}$ only has a value other than zero when there are between five and ten rows of pin fins in the entire heat exchanger. This friction factor varies as to whether V_{max} occurs at the transverse plane or the diagonal plane. V_{max} always occurs at the transverse plane for aligned pin fin arrays. Even though this represents the inlet and outlet pressure losses, it is accounted for at each zone by finding the total and dividing it up by the number of zones. This is because TEPSS solves the fin equations for each zone treating them as identical. The friction factor, $\varphi_{t,n}$, only accounts for flow at the inlet and outlet which should not be penalized for interior zones. The value is divided by the number of zones and accounted for in full when all the pressure losses are added together. Also, there is no correlation for less than five rows in the entire heat exchanger so the friction factor cannot be calculated under those conditions.

The turbulent component of the Hagen number, Hg_{turb}, for staggered pin fins can be seen in equation (5.3.89). It also uses the method in equation (5.3.88) to calculate the inlet and outlet pressure losses for the entire heat exchanger.

$$Hg_{turb} = \left\{\left[1.25 + \frac{0.6}{(X_T - 0.85)^{1.08}}\right] + 0.2\left(\frac{X_L}{X_T} - 1\right)^3 - 0.005\left(\frac{X_T}{X_L} - 1\right)^3\right\} \times Re^{1.75} + \phi_{t,n} Re^2 \quad (5.3.89)$$

The turbulent component of the Hagen number for staggered pin fins sometimes requires a correction factor for high Reynolds numbers. For $Re>250{,}000$, equation (5.3.90) is valid.

$$Hg_{turb,corr} = Hg_{turb}\left(1 + \frac{Re - 250{,}000}{325{,}000}\right) \quad (5.3.90)$$

$Hg_{turb,corr}$ is the corrected turbulent component for staggered pin fins of the Hagen number and takes the place of Hg_{turb} in equation (5.3.81) when $Re>250{,}000$. The data that the equation was developed from wasn't accurately reflected by the equations at Reynolds numbers this high.

Now that all of the components of the Hagen number have been calculated, it is possible to determine the Hagen number itself. For aligned pin fin arrays, equation (5.3.80) is used, and for staggered pin fin arrays, equation (5.3.81). The pressure drop for aligned pin fins can be calculated with the results of equation (5.3.80) as seen in equation (5.3.91).

$$\Delta P = \frac{\mu^2}{\rho}\frac{N_{rows}}{D_f^2} Hg \quad (5.3.91)$$

The pressure drop for staggered pin fins depends on where V_{max} occurs. Equation (5.3.92) shows how to calculate ΔP using the result for Hg from equation (5.3.81).

$$\Delta P = \begin{cases} \frac{\mu^2}{\rho}\frac{N_{rows}}{D_f^2} Hg & \text{when } V_{max} \text{ occurs at } A1 \\ \frac{\mu^2}{\rho}\frac{N_{rows}-1}{D_f^2} Hg & \text{when } V_{max} \text{ occurs at } A2 \end{cases} \quad (5.3.92)$$

The thermal resistance of the fins and the pressure drop across the fins has been determined for a single zone. Each zone is considered to be identical so the individual zone is expanded to represent the entire heat exchanger.

5.4 Summary

A thermoelectric heat exchanger to be used as a power recovery unit has been developed. There exists a set of equations to be used for solving a system that utilizes thermoelectric generators. With the performed energy balance completed, power recovery prediction is available. The applicability includes several heat exchanger types and set ups. Varying operating conditions are available to be analyzed by the model. Several important factors that feed into the system model are capable of being calculated or directly provided. The thermoelectric generator key parameters have several ways to be determined. These include, direct input to the system, calculation as a result of P-type and N-type leg design inputs, two methods for calculating the key parameters from thermoelectric cooler manufacturer datasheets, and a method for calculating the key parameters from the thermoelectric generator manufacturer datasheets. Three types of fins, one type having two configuration options, were analyzed. Rectangular straight fins, offset strip fins, and aligned or staggered pin fins are built and performance is calculated using sound industry accepted methods. Three dimensional heat spreading developed in Chapter 4 was also utilized in the calculations of the fin properties. A different fin type can be chosen for each side of the thermoelectric heat exchanger. How each of the calculations from this chapter is used together can be found in the next chapter describing how TEPSS functions. TEPSS incorporates the above models and their impact on performance. It interacts with optimization and solution routines to determine the total performance of the heat exchanger and any system it is used in.

6 Thermoelectric Power System Simulator (TEPSS)

Rochester Institute of Technology Sustainable Energy Lab developed the simulation software tool known as the Thermoelectric Power System Simulator, or TEPSS. A detailed overview of the TEPSS architecture followed by the application of the thermoelectric heat exchanger model described in Chapter 5 is the focus of this chapter.

6.1 Architectural Overview

The ideology behind the Thermoelectric Power System Simulator is to have a tool available for energy system models that can both solve the model as well as optimize parameters within the model. The architecture for TEPSS was created by Kreuder [52]. The MATLAB® programming software is chosen for the development of TEPSS for several reasons. MATLAB® is both widely used by engineers in all disciplines and contains an add-on package known as the Optimization Toolbox. Optimization Toolbox is a powerful set of functions with extensive ability to provide optimal solutions.

TEPSS uses object oriented programming which allows variables to be assigned to a class. An instance of a class is called an object. Objects contain data structures known as *properties* and a set of functions known as *methods*. Each object contains all the necessary data and algorithms to operate. Values for variables are applied through a script and these variables are distributed throughout the object allowing for simulation of that specific object. An object can also store other objects within itself.

For TEPSS, objects can be component objects or node objects. Component objects can be a physical device, or component, in an energy system which includes pumps, turbines, generators, condensers, heat exchangers, etc. Node objects, or nodes, store dependent variables which are solved for during the simulation. Nodes provide the interface between component objects meaning that the properties of a node are shared between connected components. Nodes can be defined for various domains depending on the dependent variables stored inside the defined domain. For example, the fluid domain contains dependent variables mass flow, specific enthalpy, and pressure; and the electrical domain contains dependent variables voltage and current. In the case of the fluid domain, the user inputs an absolute temperature in place of the specific enthalpy. Specific enthalpy is then derived from the thermodynamic state variables through *FluidProp* [53], an activex server that runs in the background of TEPSS. Figure 6.1 shows the interconnection between component objects and node objects.

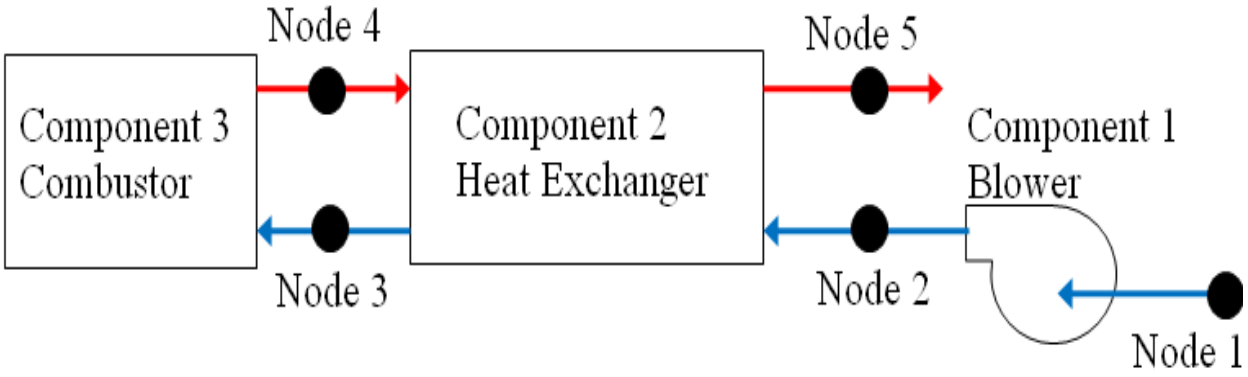


Figure 6.1: Component objects connected by fluid nodes [52].

To make the TEPSS architecture functional, the user must define a plethora of inputs that define the energy system. Component and node configuration, initial guesses, boundary conditions, and component parameter settings must all be defined in order to obtain a system solution. In order to optimize the system, more information is required from the user. This information includes design variables, constraints, and convergence criteria.

Several constructor functions are run to create all of the components and link them to their respective nodes. Additional functions are implemented to set up the boundary conditions and initial guesses. After the assembly of the system is complete, the solver constructor checks to ensure that the system is solvable. The number of node variable guesses equal to the total number of equations in the system will result in a square matrix, thus potentially solvable.

The TEPSS code then uses Newton's Method and the Jacobian matrix to arrive at a steady state solution. The Jacobian matrix is a square matrix containing the first partial derivatives of the system equations with respect to each of the unknowns. Design variables can be optimized using TEPSS once a system solution is converged upon. The optimization portion of TEPSS is outside the scope of this thesis but is a highly recommended tool for development and incorporation of a thermoelectric heat exchanger into a thermodynamic system. More detailed information on TEPSS can be found in Kreuder's thesis [52].

6.2 Thermoelectric Power Unit System Model in TEPSS

The thermoelectric power unit system model developed in Chapter 5 is implemented within the Thermoelectric Power System Simulator. Using the architecture set forth by Kreuder [52], the object *tepowerunit* was created to incorporate the component model. TEPSS is capable of being expanded to any type of thermodynamic system component if the object oriented programming format is followed. The challenge of the expansion of the *tepowerunit* model into TEPSS was merging the model with the software architecture to allow use in the simulation and optimization

routines of TEPSS. The model is far more complex than the existing TEPSS components and required in depth implementation.

The object *tepowerunit* code can be seen in Appendix A. It follows the general principles of object oriented programming and contains a data structure known as *properties* and a set of functions known as *methods*. Within *properties*, more specific parameters are kept inside structures for the thermoelectric module properties, heat exchanger fin properties, heat exchanger unit properties, and cost related properties to mention a few. In addition, much of the heat flow values are kept for ease of access by the object methods.

Table 6.1 lists the parameters that can be assigned to the system depending on the selections made by the user that defines the method of determination of module parameters. Table 6.2 displays the inputs for the type of heat exchanger that define the unit. Table 6.3 shows the inputs for the type of fins for the hot and cold side of the heat exchanger as well as the defining dimensions. With all of the necessary properties provided, *tepowerunit* must solve the many contained *methods* within. All of the inputs can be seen in the script in Appendix B which is the script used to solve TEPSS and compare the analytical results to the experimental results obtained which are discussed in Chapter 7.

Inside of *methods* are the engineering models for the thermoelectric power unit. Various *functions* exist and are activated depending on what options are selected via user input. Within these functions, there are also logic statements determining which portions of the function need to be run based on the input from the user. Options include the way in which the thermoelectric module parameters are determined based on the five methods described in Sections 5.2.1 – 5.2.5; the type of heat exchanger from either of Sections 5.1.4 or 5.1.5 in which counter flow, isothermal, or constant heat rate may be selected; the type of fins on the hot side of the heat exchanger from Sections 5.3.1 – 5.3.3 in which rectangular straight fins, offset strip fins, aligned pin fins, or staggered pin fins are options; and the type of fins on the cold side of the heat exchanger which have the same options as the hot side but have the ability to be of a different geometry. Based on the selections made by the user, the heat exchanger unit is set up, the thermoelectric module parameters are defined, and the hot side fins and cold side fins, if applicable, are created and applied to the system. The thermal resistance for the fins takes into account the heat spreading using the model developed in Section 4.2. A parameter check method is run to ensure no unrealistic features are created such as a negative physical dimension. If such a feature exists, the method will end the simulation and recommend the user adjust parameter settings.

At this point, the thermoelectric power unit has been built by the software but thermodynamic simulations have yet to occur. The component object is represented through each of the parameters defined from the initial input. The *UA*-value from equation (5.3.39) is actually found for the hot side and cold side, when applicable, through the *compute* method because it is dependent on the thermodynamic properties. Each of the thermodynamic states is defined with

regards to the nodes associated with the power unit. The compute function then defines the number of zones and calls the *fsolve* method. *fsolve* is a MATLAB® tool used for solving systems of nonlinear equations. The parameters that define the thermoelectric modules, the heat exchanger geometry, and the fin thermal properties are fed into the set of equations being solved by *fsolve*. This set of equations comes from Section 5.1.4, counter flow power unit, or 5.1.5, isothermal or constant heat rate power unit. *fsolve* computes an answer for each of the unknown

Table 6.1: User controlled simulation variables for thermoelectric modules.

*Variable**	*Parameter*
All Module Input Options	
module.zone_load_res	Modules electrical load resistance per zone
Module Inputs - Section 5.2.1, Section 5.2.3, Section 5.2.4, & Section 5.2.5	
module.width	Module width (assumed to be square)
module.thickness	Module thickness
Module Inputs - Section 5.2.1	
module.talpha	Module Seebeck coefficient
module.ttherm_resist	Module thermal resistance
module.telec_resist	Module electrical resistance
Module Inputs - Section 5.2.2	
module.rho_p	Electrical resistivity of p-type semiconductor
module.alpha_p	Seebeck coefficient of p-type semiconductor
module.k_p	Thermal conductivity of p-type semiconductor
module.l_p	Length of p-type semiconductor legs
module.area_p	Leg cross sectional area for p-type semiconductor
module.rho_n	Electrical resistivity of n-type semiconductor
module.alpha_n	Seebeck coefficient of n-type semiconductor
module.k_n	Thermal conductivity of n-type semiconductor
module.l_n	Length of n-type semiconductor legs
module.area_n	Leg cross sectional area for n-type semiconductor
module.a_ratio	Ratio of module area to semiconductor cross sectional area in a module
module.num	Number of leg pairs in a module
module.l_cer	Module ceramic thickness
module.k_cer	Module ceramic thermal conductivity
module.contact_resist	Electrical contact resistance in the module
Module Inputs - Section 5.2.3 & Section 5.2.4	
module.T_h	TEC data sheet reported module hot side temperature
module.deltaTmax	TEC data sheet reported module maximum temperature difference
module.Imax	TEC data sheet reported maximum current
module.Vmax	TEC data sheet reported maximum voltage, option 3 only
module.Qmax	TEC data sheet reported maximum heat absorbed at the cold face, option 4 only
Module Inputs - Section 5.2.5	
module.V_oc	TEG data sheet reported open circuit voltage
module.T_h	TEG data sheet reported module hot side temperature
module.T_c	TEG data sheet reported module cold side temperature
module.I_load	TEG data sheet reported load current
module.q_h	TEG data sheet reported module hot side heat rate

*Each variable input into TEPSS starts with parameters.tepowerunit.

Table 6.2: User controlled simulation variables for thermoelectric power unit.

*Variable**	*Parameter*
unit.num	Number of in line finite elements (zones)
unit.series	Number of modules in the length of the power unit per zone
unit.parallel	Number of modules across the width of the power unit
unit.insul_k	Thermal conductivity of bypass leakage insulation
unit.zone_to_mod_area_ratio	Ratio of total heat exchanger flux area to module area
unit.therm_contact_res	Thermal contact resistance between power unit fins and module
unit.uvalue	Overall heat transfer coefficient to environment
unit.shell_t	Thickness of power unit walls (insulation)
unit.shell_k	Conductivity of power unit walls (insulation)
unit.envir_temp	Surrounding environment temperature
options	Power unit settings

*Each variable input into TEPSS starts with parameters.tepowerunit.

Table 6.3: User controlled simulation variables for thermoelectric power unit fins.

*Variable**	*Parameter*
All Fin Types	
fins.k_h	Hot side fin thermal conductivity
fins.base_t_h	Thickness of the base of the hot side fin array
fins.l_h	Hot side fin length
fins.k_c	Cold side fin thermal conductivity
fins.base_t_c	Thickness of the base of the cold side fin array
fins.l_c	Cold side fin length
Rectangular Straight Fins - Section 5.3.1	
fins.t_h	Hot side fin thickness
fins.num_h	Hot side number of fins
fins.t_c	Cold side fin thickness
fins.num_c	Cold side number of fins
Offset Strip Fins - Section 5.3.2	
fins.t_h	Hot side fin thickness
fins.trans_num_h	Hot side number of fins in the transverse direction
fins.rows_num_h	Hot side number of rows of fins
fins.t_c	Cold side fin thickness
fins.trans_num_c	Cold side number of fins in the transverse direction
fins.rows_num_c	Cold side number of rows of fins
Aligned and Staggered Pin Fins - Section 5.3.3	
fins.dia_h	Hot side pin fin diameter
fins.pitch_trans_h	Hot side transverse pin fin pitch
fins.pitch_long_h	Hot side longitudinal pin fin pitch
fins.dia_c	Cold side pin fin diameter
fins.pitch_trans_c	Cold side transverse pin fin pitch
fins.pitch_long_c	Cold side longitudinal pin fin pitch

*Each variable input into TEPSS starts with parameters.tepowerunit.

variables inside the equation set and the variables are then fed back into the *compute* method. This is repeated for the next zone of the heat exchanger until all zones have been solved. Figure 5.3 shows the variables solved for each zone with *j* representing the first zone. *j+1* represents the next zone and the figure displays which temperature variables are carried over for the simulation of the subsequent zone.

The values determined through the *fsolve* function are used to evaluate convergence criterion or residuals. The convergence criterion is sorted into an array and interacts with the node objects for all components of a proposed system. In the case of an isolated thermoelectric power unit, the system attempts to converge toward the defined boundary conditions. These are provided by the absolute temperatures of the heat exchanger at each of the nodes, the mass flow rate, and the pressure drop across the device. Six convergence equations are shown in (6.2.1)-(6.2.6) as they make up an array for the counter flow thermoelectric heat exchanger.

$$e(1) = \dot{m}_{h,in} - \dot{m}_{h,out} \tag{6.2.1}$$

$$e(2) = \dot{m}_{c,in} - \dot{m}_{c,out} \tag{6.2.2}$$

$$e(3) = P_{h,in} - P_{h,out} - \Delta P_{\mathrm{h}} \tag{6.2.3}$$

$$e(4) = P_{c,in} - P_{c,out} - \Delta P_{\mathrm{c}} \tag{6.2.4}$$

$$e(5) = \dot{m}_{h,in}\left(h_{h,in} - h_{h,out}\right) - q_h - q_{loss,h} - q_{ins} \tag{6.2.5}$$

$$e(6) = \dot{m}_{c,in}\left(h_{c,in} - h_{c,out}\right) - q_c + q_{loss,c} - q_{ins} \tag{6.2.6}$$

Equations (6.2.1) and (6.2.2) represent conservation of mass flow on the hot side and cold side of the heat exchanger, respectively. Equations (6.2.3) and (6.2.4) show the conservation of pressure on the hot side and cold side of the heat exchanger, respectively. Equations (6.2.5) and (6.2.6) display the conservation of energy on the hot side and cold side, respectively. Convergence equations for an isothermal or constant heat rate heat exchanger are similar to the ones shown here but less in number. They can be deduced from the *tepowerunit* computer code in Appendix A.

After the zones are stepped through once, the residual values have been defined and are reported back to the TEPSS simulation software. These values are then compared against the defined convergence set point to determine if the solution has reached steady state. If a steady state solution has been reached, then the component is solved for and a solution is provided. An optimized solution can be sought after as well if the initiating script prompted for one. If a steady state solution has not been reached, then the *compute* method is run again with a new set of node states in an attempt to get closer to a system solution. This iterative process repeats until the convergence criteria defined by the user are met. Then, the costing function is evaluated.

6.3 Thermoelectric Power Unit Costing Function

The *tepowerunit* component has a built-in component cost function. The component cost function is evaluated to supply values to the overall TEPSS costing function. This is used for optimization because it can define the design variables to move toward a preferred design and solution. The costing function only serves a purpose when the optimization routine is invoked as it has no purpose when solving for the system steady state solution. It is described here in detail to provide information in support of Kreuder's work [52].

The costs of the modules are determined in a couple different ways. If the modules key parameters are directly provided or the module parameters are determined via manufacturer data, then the cost of the modules is the cost of an individual module times the cost of the total number of modules used. If the module key parameters are calculated from p-type and n-type semiconductor leg geometries and material properties, then several calculations must be made. The leg material volume is required for both the p-type and n-type legs. Equation (6.3.1) shows the method for finding the volume for the p-type legs in a thermoelectric module, V_p. N represents the number of leg pairs.

$$V_p = L_p A_p N N_{mod,zone} N_z \tag{6.3.1}$$

The volume of leg material, V_p, is used to obtain the cost of the p-type legs by multiplying it by the cost of the leg material per unit volume provided by the user. Equation (6.3.1) can be used for the n-type legs with the same method applied for obtaining the cost. The ceramic volume is also calculated and multiplied by the cost of ceramic per unit volume. The values for the p-type leg cost, n-type leg cost, and ceramic cost are all added to any additional costs per leg pair times the number of modules and the total manufacturing cost. This provides the total finances for all of the thermoelectric modules in the system.

A similar approach is taken for finding the cost of the hot and cold side fin geometries. A volume of the hot side fins is found and multiplied by the cost of the fin material per unit volume. Each type of fins has its own equation for finding volume. An associated manufacturing cost is provided for the fin set which is then added to the material cost. This approach is repeated for the cold side fins if the proposed system has them. The total cost of the fins is the sum of the hot side and cold side fins material and manufacturing costs.

Costs for assembly take into account anything that has not been previously accounted for. The volume of insulation used to improve performance of the heat exchanger is calculated. This is multiplied by a cost per unit volume to obtain the total cost of the insulation. Any other costs per zone are taken into account as well as any fixed costs associated with the power unit. Finally, a fixed assembly cost is built into this portion of the costing function.

$$Total\ Cost = \sum Module\ Costs + \sum Heat\ Sink\ Costs + \sum Assembly\ Costs \tag{6.3.2}$$

All of the aforementioned costs are summed together as seen in equation (6.3.2) and built into a matrix that TEPSS can utilize for optimization. Table 6.4 shows the associated parameters that are required by the user to input into the run-script. The thermoelectric power unit has a negative cost or a costing gain in that it produces DC electricity through its heat recovery from the thermoelectric modules. This is also factored into the costing function causing the unit to potentially have a positive impact over an extended period of time.

As previously mentioned, the costing function provides feedback into TEPSS for the optimization routine. The design inputs displayed in Table 6.1, Table 6.2, and Table 6.3 are held constant throughout the analysis unless the input is provided as a design variable. If any inputs are chosen to be design variables, they will be adjusted through the optimization routine after consecutive simulations to improve the costing function. The design variable is provided with upper and lower bounds determined by the user. TEPSS can be run with multiple design variables and the optimization routine is run until a best result is determined.

Table 6.4: User controlled simulation variables for costs associated with the thermoelectric power unit.

*Variable**	*Parameter*
cost.specific.module	Cost per module**
cost.specific.module_p_leg_material	Cost per cubic meter of p-type material; module inputs - Section 5.2.2
cost.specific.module_n_leg_material	Cost per cubic meter of n-type material; module inputs - Section 5.2.2
cost.specific.module_ceramic	Cost per cubic meter of ceramic material; module inputs - Section 5.2.2
cost.specific.cost_per_leg_pair	Additional cost per thermoelectric leg pair; module inputs - Section 5.2.2
cost.fixed.module_manufac	Manufacturing cost of modules; module inputs - Section 5.2.2
cost.specific.fin_material	Cost per cubic meter of fin material
cost.fixed.fin_manufac_h	Manufacturing cost of hot side fins
cost.fixed.fin_manufac_c	Manufacturing cost of cold side fins
cost.specific.insulation	Cost per cubic meter of insulation
cost_per_zone_area	Cost per square meter of heat transfer area
cost.fixed.other	Miscellaneous fixed costs
cost.fixed.assembly	Power unit assembly cost

*Each variable input into TEPSS starts with parameters.tepowerunit.
**Used for module inputs from Section 5.2.1, Section 5.2.3, Section 5.2.4, & Section 5.2.5

6.4 Summary

The model for the thermoelectric power unit has been developed into a software component for simulations using Rochester Institute of Technology Sustainable Energy Lab's Thermoelectric Power System Simulator. Appendix A contains the code for *tepowerunit* in a functional, object-oriented program. Appendix B contains a script for running *tepowerunit* while utilizing the TEPSS simulator and optimization package developed by Kreuder [52]. This combination of the TEPSS computer software provides insight into the feasibility of implementing a thermoelectric

power unit heat exchanger into an actual heat recovery platform. The *tepowerunit* can be simulated individually to determine the capabilities of a proposed or existing device. Various parameters of the *tepowerunit* can be optimized in a system to determine the cost effectiveness and pay back of such a device. With the TEPSS software architecture, it is possible to expand or manipulate *tepowerunit* to incorporate newly developed thermoelectric heat exchanger models. This allows for a performance comparison across similar models throughout the research community.

7 Experimental Results for the Thermoelectric Power Unit System Model

7.1 Experiment Setup

The thermoelectric power unit test setup utilized the same test bed as the heat spreading test setup. The heat exchanger is adjustable and allowed for the aluminum blocks to be set between the hot and cold sides as well as thermoelectric modules. Figure 7.1 shows the test heat exchanger and instrumentation for the thermoelectric modules. Data is taken by the same thermocouples as described in Section 4.4. Steady state is reached in the same manner as described previously.

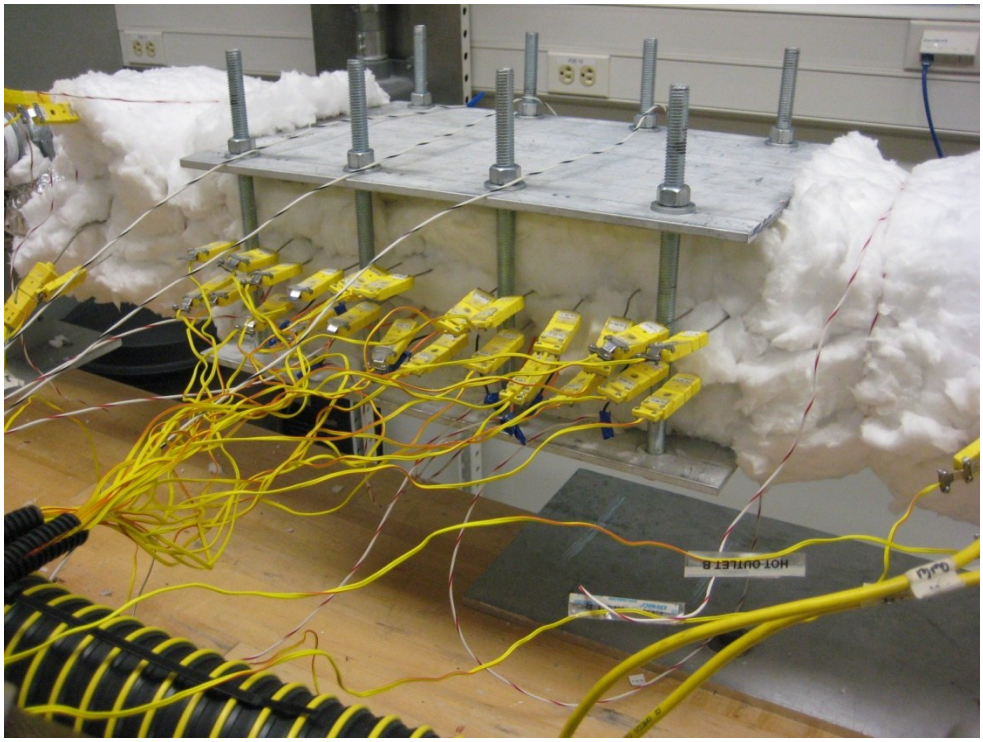


Figure 7.1: Thermoelectric power unit test setup picture.

Twelve different operating tests are performed for the thermoelectric power unit test setup. The varying temperature conditions include inline heater temperature set points of 100°C, 150°C, 200°C, and 235°C. The varying flow rate conditions include 50 SCFM (1.42 m^3/min), 75 SCFM (2.12 m^3/min), and 100 SCFM (2.83 m^3/min).

The thermoelectric modules are arranged into three zones with four modules per zone for a total of twelve modules in the thermoelectric power unit. The modules are connected electrically in series in each zone as shown in Figure 7.2. Each string of modules is connected to a variable load resistor which is dialed in to a set resistance for each zone. When load resistance equals module

internal electrical resistance, the maximum power point is obtained. Maximum power point would have been obtained but the capabilities to set each zone load resistance to match internal electrical resistance did not exist. The intent was to have each zone equal to one another with regards to load resistance to enable simplicity in simulations. Therefore, load resistance was approximately matched for each zone and relatively close to the maximum operating point. The average load resistance for each test is shown in Table 7.1.

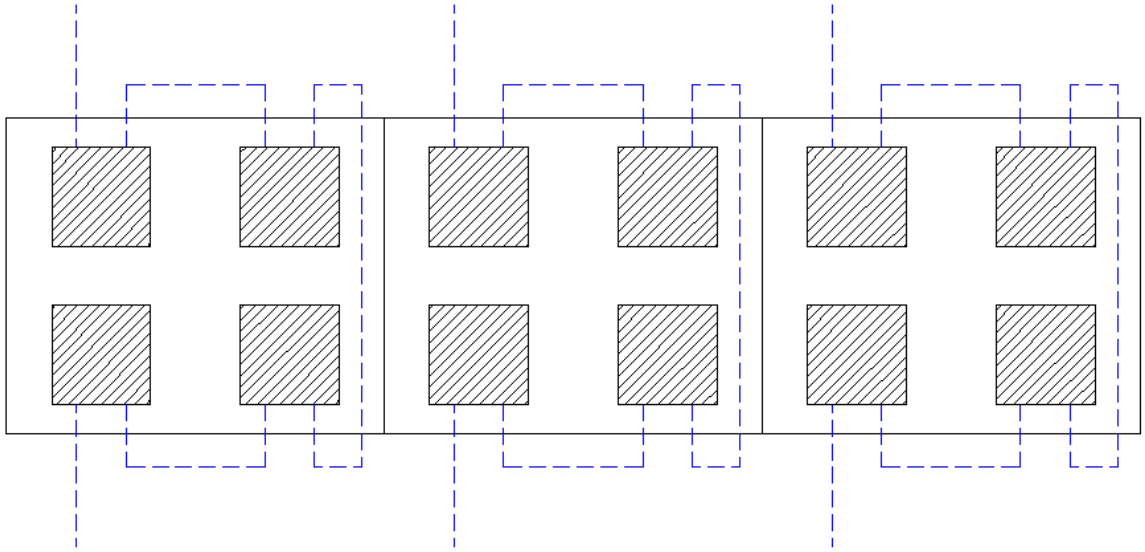


Figure 7.2: Thermoelectric module configuration for thermoelectric power unit test setup.

Table 7.1: Load resistance for each test.

Set Point Temperature	Flow Rate	Load Resistance
[°C]	[SCFM]	[Ω]
235	100	7.9
235	75	7.8
235	50	7.6
200	100	7.9
200	75	7.8
200	50	7.5
150	100	8.3
150	75	7.8
150	50	7.5
100	100	8.2
100	75	8.1
100	50	8.2

Data was collected and reported in the following section. Simulations were performed based on the measured boundary conditions reported in Table 7.2.

Table 7.2: Simulation boundary conditions measured during data collection for each test.

Set Point Temperature	Flow Rate	Mass Flow Rate	Tc,in	Tc,out	Th,in	Th,out
[°C]	[SCFM]	[kg/s]	[°C]	[°C]	[°C]	[°C]
235	100	0.055	45.5	67.5	247.0	219.0
235	75	0.043	39.0	67.5	252.2	218.3
235	50	0.029	39.9	77.3	247.9	205.9
200	100	0.056	46.2	63.7	210.1	187.6
200	75	0.043	39.1	62.1	212.8	185.0
200	50	0.029	40.1	69.9	209.9	175.8
150	100	0.057	45.5	57.2	157.5	142.2
150	75	0.041	37.7	54.3	158.6	138.7
150	50	0.029	38.8	59.2	155.8	132.0
100	100	0.058	44.9	50.9	103.3	95.7
100	75	0.042	37.5	46.2	103.6	93.2
100	50	0.030	38.0	48.9	102.4	89.9

7.2 Initial Results

A simulation using the model developed in earlier chapters was performed for each of the experimental tests with the input parameters set to the values shown in Table 7.3. The values provided are known based on the design of the test bed thermoelectric power unit. It should be noted that the base thickness of the fins on the cold is thicker than that of the hot side. This is because an aluminum spacer plate was added to enable the heat exchanger to provide ample contact pressure across all modules by mitigating interference issues. The value for the thermal contact resistance was determined from the data from the heat spreading tests. Module parameters were determined through characterization testing using the test stand described in section 3.4.1. Characterization tests were performed at hot side temperatures of 150°C. Modules that most closely resembled one another were used for the power unit experiments to provide average values to be used as the inputs into the TEPSS simulation. The same fin geometries used in the heat spreading tests were used in the module testing phase.

Table 7.3: Initial Simulation Input Parameters.

*Variable**	*Parameter Value*	*Parameter Unit*
unit.num	3	[-]
unit.series	2	[-]
unit.parallel	2	[-]
unit.insul_k	0.06	[W/(m-K)]
unit.zone_to_mod_area_ratio	3	[-]
unit.therm_contact_res	5.45E-5	[(K-m^2)/W]
unit.uvalue	5	[W/(m^2-K)]
unit.shell_t	0.0254	[m]
unit.shell_k	0.3	[W/(m-K)]
unit.envir_temp	300	[K]
module.width	0.04	[m]
module.thickness	0.0033	[m]
module.talpha	0.0460	[V/K]
module.ttherm_resist	1.5515	[K/W]
module.telec_resist	2.422	[Ω]
fins.l_h	0.0254	[m]
fins.t_h	0.001905	[m]
fins.num_h	15	[-]
fins.k_h	209	[W/(m-K)]
fins.base_t_h	0.012319	[m]
fins.l_c	0.0254	[m]
fins.t_c	0.001905	[m]
fins.num_c	15	[-]
fins.k_c	209	[W/(m-K)]
fins.base_t_c	0.021844	[m]

*Each variable input into TEPSS starts with parameters.tepowerunit.

The experimental results measured via testing and initial model simulation results can be found in Table 7.4. The measured and initial model results are also graphed in Figure 7.3 with regards to power versus hot side inlet temperatures. A percent difference plot for the initial results displaying the percent difference between the measured power, P_{meas}, and modeled power, P_{model} is shown in Figure 7.4. Percent difference is represented by equation (7.2.1).

$$\%\, Difference = \left(\frac{P_{model} - P_{meas}}{P_{meas}}\right) \times 100 \qquad (7.2.1)$$

Table 7.4: Measured power and modeled power initial results.

Temperature [°C]	Flow Rate [SCFM]	Measured Power [W]	Modeled Power [W]
235	100	13.49 +/- 0.24	22.74
235	75	12.75 +/- 0.23	20.80
235	50	9.06 +/- 0.16	13.34
200	100	10.25 +/- 0.19	15.24
200	75	9.71 +/- 0.18	13.84
200	50	7.00 +/- 0.13	9.02
150	100	5.36 +/- 0.10	7.31
150	75	5.22 +/- 0.10	6.46
150	50	3.82 +/- 0.07	4.25
100	100	1.82 +/- 0.04	2.02
100	75	1.87 +/- 0.04	2.01
100	50	1.37 +/- 0.03	1.38

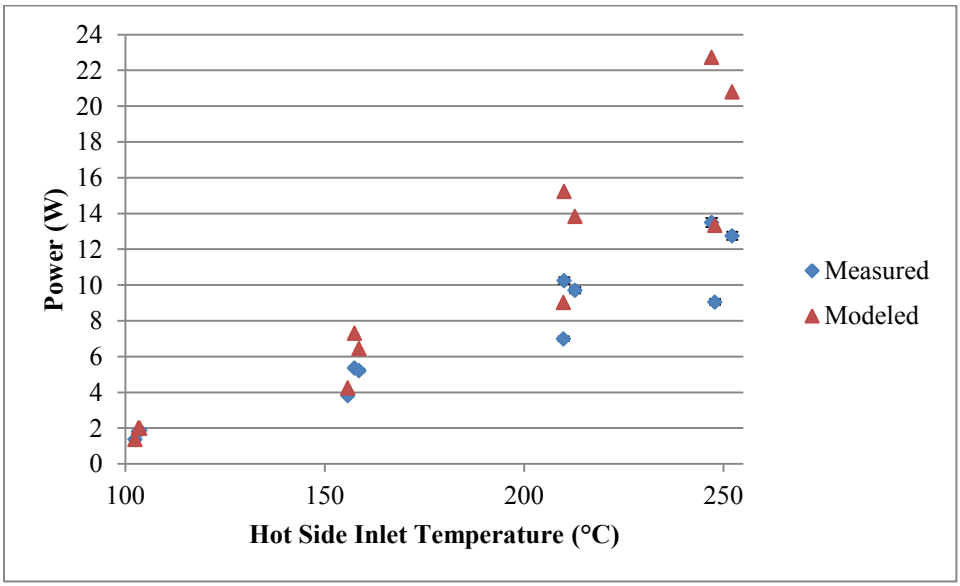


Figure 7.3: Measured power and modeled power initial results.

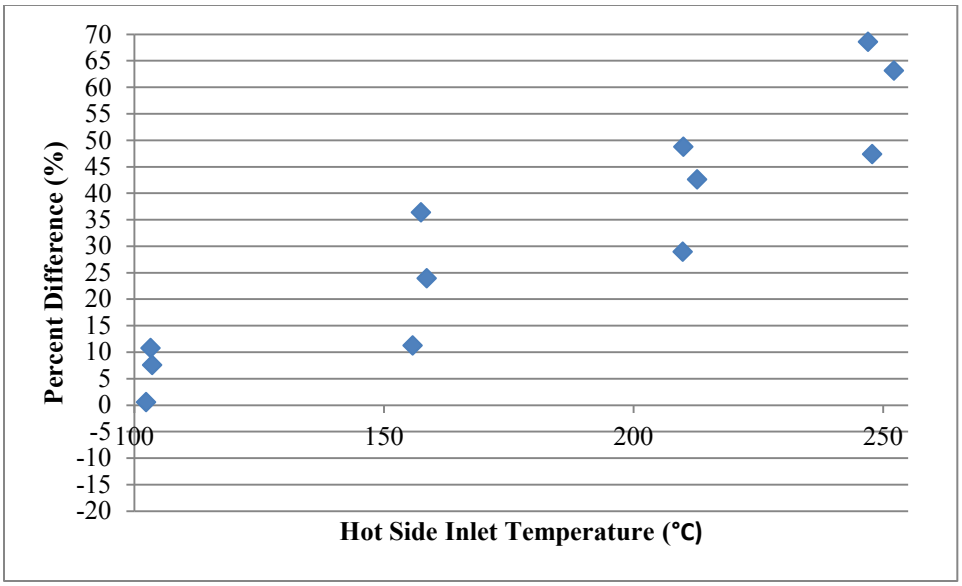


Figure 7.4: Measured power and modeled power initial results percent difference.

An uncertainty analysis is performed to determine if the simulated results fall within the range of uncertainty of the measurements. This uncertainty analysis is only based on the precision and random uncertainty. It does not account for the systematic or biased error such as an incorrect measurement or drift. Uncertainty for the measurements taken with regards to power must consider measured load voltage, shunt voltage, and shunt resistance. A shunt resistor was placed in series with the load resistor. This allowed for the current to be determined so that ultimately power could be found. The measured power was calculated using equation (7.2.2).

$$P = \frac{V_{L,1}V_{S,1}}{R_S} + \frac{V_{L,2}V_{S,2}}{R_S} + \frac{V_{L,3}V_{S,3}}{R_S} \tag{7.2.2}$$

V is the voltage and R is the resistance while subscript L represents the electrical load and subscript s represents electrical values related to the shunt resistor. Numerical subscripts signify the zone number. The uncertainty in the power is the square root of the sum of the three zones power calculated with the uncertainty in the individual measurements accounted for as seen in (7.2.3).

$$\Delta P = \sqrt{\sum_{i=1}^{3}\left[\left(\frac{\Delta V_{L,i}V_{S,i}}{R_{S,i}}\right)^2 + \left(\frac{V_{L,i}\Delta V_{S,i}}{R_{S,i}}\right)^2 + \left(\frac{V_{L,i}V_{S,i}\Delta R_{S,i}}{{R_{S,i}}^2}\right)^2\right]} \tag{7.2.3}$$

Rearranging equation (7.2.3) to account for the percent uncertainty results in equation (7.2.4).

$$\frac{\Delta P}{P} = \sqrt{\sum_{i=1}^{3}\left[\left(\frac{\Delta V_L}{V_{L,i}}\right)^2 + \left(\frac{\Delta V_S}{V_{S,i}}\right)^2 + \left(\frac{\Delta R_S}{R_S}\right)^2\right]} \tag{7.2.4}$$

The DAQ card is 12-bit so the precision is ½ raised to 12. This means that ΔV_L equals 4.88 mV for a +/-10 V scale and 2.44 mV for a +/-5 V scale. The +/-10 V scale was used for each of the 200°C and 235°C tests and the +/-5 V scale was used for all of the 150°C and 100°C tests. The shunt voltage was measured on a +/-100 mV scale so the uncertainty in those measurements is +/-48.8 μV. The reported value of uncertainty for the shunt resistor, $\Delta R_s/R_s$, is 1%. Each of the three thermoelectric power unit zone uncertainties are added together and the square root is taken. The percent uncertainty for each of the tests is shown in Table 7.5. These results are reflected in the plus/minus values in Table 7.4 and with error bars in Figure 7.3.

Table 7.5: Percent uncertainty for the measured power of each test.

Temperature	Flow Rate	Uncertainty
[°C]	[SCFM]	[%]
235	100	1.8
235	75	1.8
235	50	1.8
200	100	1.8
200	75	1.8
200	50	1.9
150	100	1.9
150	75	1.9
150	50	1.9
100	100	2.2
100	75	2.2
100	50	2.4

As can be seen in the initial results table and figures, discrepancies between the measured and modeled power are greater than precision uncertainty. Further analysis is required to determine the reason for the discrepancies.

7.3 Simulation Adjustments

7.3.1 Temperature Dependent Properties

All of the thermoelectric modules used in this phase of testing were characterized at 150°C in the test stand described in section 3.4.1. There was concern that the module key parameters could vary based on operating temperatures. Looking at testing previously performed by Sandoz-Rosado, it was apparent that the module internal electrical resistance varied significantly with operating temperature while the thermal resistance and module level Seebeck coefficient were relatively constant [15]. Over the range of 150°C to 235°C, the internal electrical resistance

increased by 20 percent. Therefore, for test cases of 235°C, the internal electrical resistance was set to 1.2 times the value reported in Table 7.3. The scale in the variation in the parameter appeared to be linear as determined from examination of data from Sandoz-Rosado [15]. Thus, 200°C were considered to have an internal electrical resistance of 1.1 times the previously reported input parameter. New results for measured power and modeled power with temperature dependent properties accounted for are shown in Figure 7.5. Figure 7.6 shows the percent difference between the measured and modeled powers. As can be seen through comparison with Figure 7.4, the percent error levels off for the higher temperature set point tests. Error is still significant for most tests and up to approximately 40% for the 235°C and 100 CFM test. Further analysis is required and additional adjustments to the simulation need to be made per findings through analysis. All the remaining analysis utilizes the adjustments to the base case for the temperature dependent internal electrical resistance.

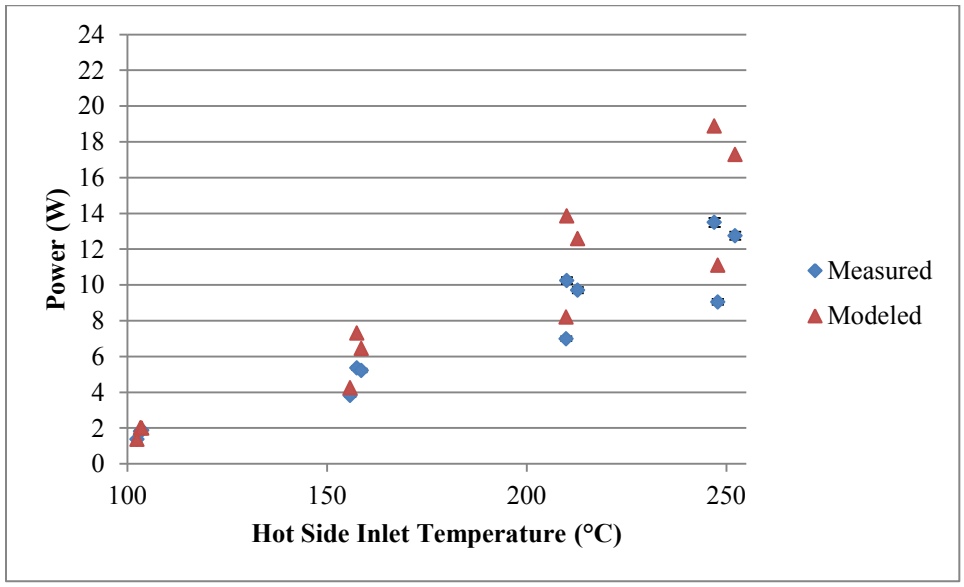


Figure 7.5: Measured power and modeled power with temperature dependent module electrical resistance results.

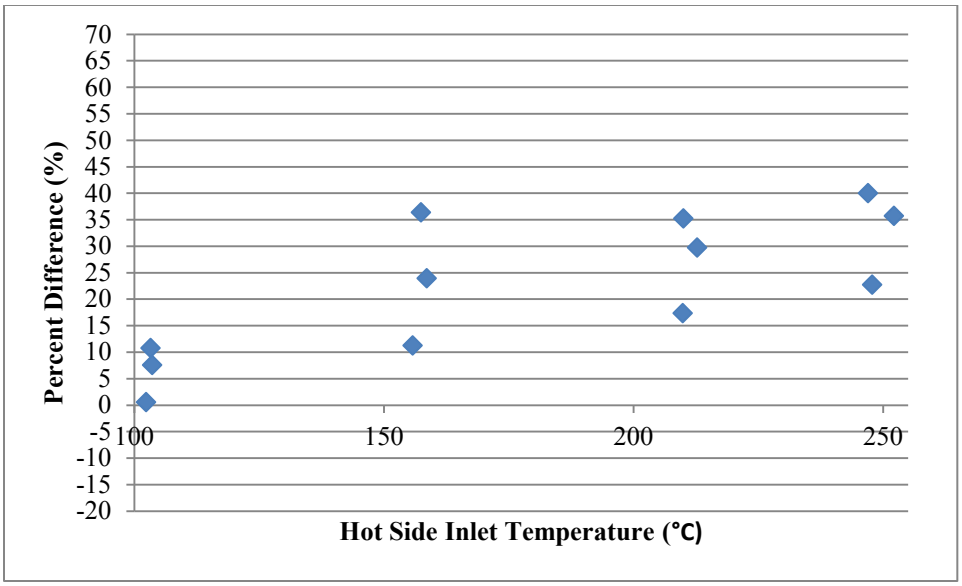


Figure 7.6: Measured power and modeled power with temperature dependent module electrical resistance percent difference.

7.3.2 Heat Losses to Surrounding Environment

The heat lost to the environment is of concern to the analysis. The model uses equations from section 5.1.4 to determine $q_{losses,h}$ and $q_{losses,c}$ and the effect of these values on the system. The *UA* values for the losses assume natural convection on the outer surfaces. The input parameter for the free convection is assumed to be 5 W/(m^2-K) which is a reasonable assumption but may provide some inaccuracy. The insulation surrounding the heat exchanger is one inch thick with an assumed conduction coefficient because the manufacturer did not have a reported value. The assumption likely has some error in it. The heat exchanger was well wrapped but heat could potentially escape at the manifolds or in the regions where the heat exchanger was not as well covered. Also, the test stand system is quite small which means edge effects are critical. Because of one dimensional analysis where the edge effects are neglected, the model likely does not fully capture the thermal losses. In a larger system, edge effects would more likely be negligible meaning better heat lost predictions. The values provided by the initial input were low when compared to the expected losses based on the measured data. Equations (7.3.1) and (7.3.2) are used to determine the amount of heat lost based on the experiment and the simulation, respectively.

$$q_{losses,meas} = \dot{m}_h C_{p,h}\left(T_{h,i} - T_{h,o}\right) + \dot{m}_c C_{p,c}\left(T_{c,i} - T_{c,o}\right) - P \qquad (7.3.1)$$

$$q_{losses,model} = q_{losses,h} + q_{losses,c} \qquad (7.3.2)$$

$q_{losses,meas}$ and $q_{losses,model}$ are the total heat exchanger losses for the measured data and the simulated results, respectively. $q_{losses,meas}$ did not equal $q_{losses,model}$ for any of the experiments

performed. $q_{losses,meas}$ was two to nine times greater than $q_{losses,model}$ for 50 CFM to 100 CFM tests respectively.

To exam the effects of the potentially higher amounts of heat lost, input parameters were adjusted to simulate nine times the heat lost in the base case. The base cases ranged from 9 W (50 CFM, 100°C) – 27 W (100 CFM, 235°C) lost heat while the nine times losses simulated cases ranged from 83 W (50 CFM, 100°C) – 232 W (100 CFM, 235°C) lost heat. The modeled power results can be seen in Figure 7.7. Figure 7.8 shows the percent difference for the adjustments made to the model input parameters. Data is not reported for the 235°C and 50 CFM test because *FluidProp* has a bad range of data that trip up the TEPSS package. *FluidProp* outputs negative or undefined values for enthalpy at the bad data points which are relatively close to the operating range for the 235°C and 50 CFM test. Unfortunately, simulated results could not be found for this test under these input conditions. Overall, the data sees only minor improvements. However, the losses are inaccurately calculated based on the measured data and should be compensated for regardless of limited improvement to the error reduction. Nine times the losses is an extreme case whereas five times losses to the environment is more likely and works well for most tests based on the analysis of the data. The cases with five times losses ranged from 46 W (50 CFM, 100°C) – 130 W (100 CFM, 235°C). Figure 7.9 and Figure 7.10 show the results for five times the base case losses. The overall effect of the inaccurate losses prediction is fairly negligible in determining the power output. Additional analysis is needed to determine the remaining discrepancies in the data.

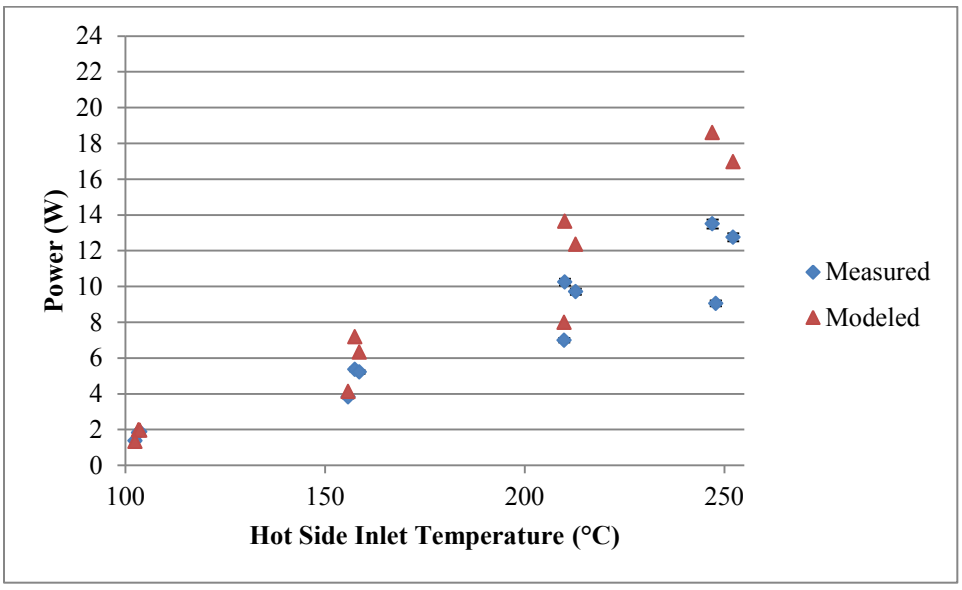


Figure 7.7: Measured power and modeled power with 9 times environmental losses results.

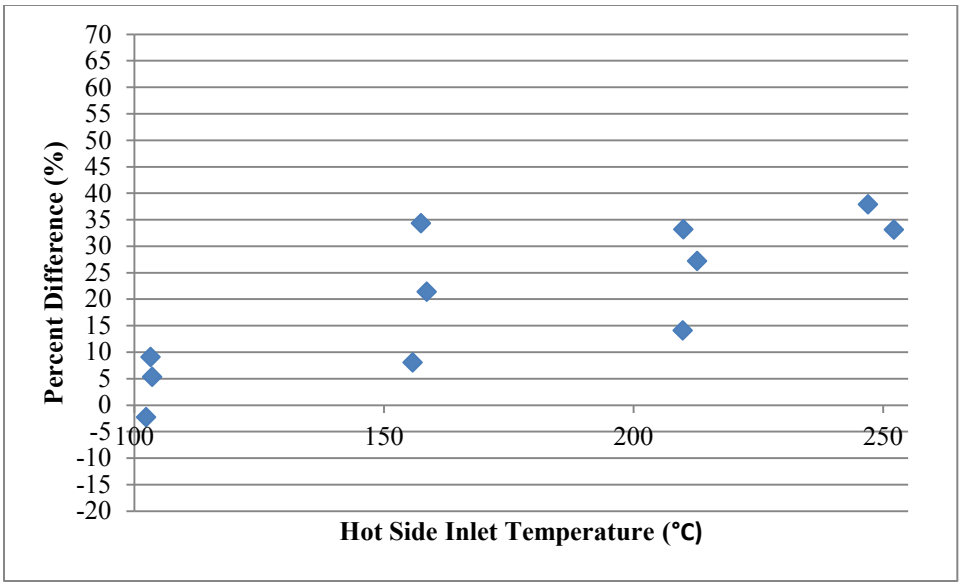


Figure 7.8: Measured power and modeled power with 9 times environmental losses percent difference.

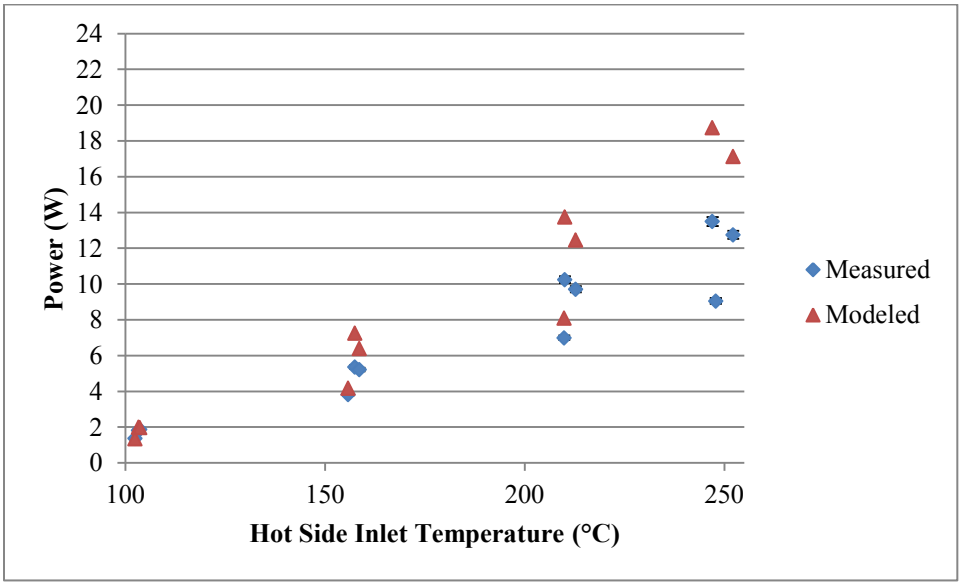


Figure 7.9: Measured power and modeled power with 5 times environmental losses results.

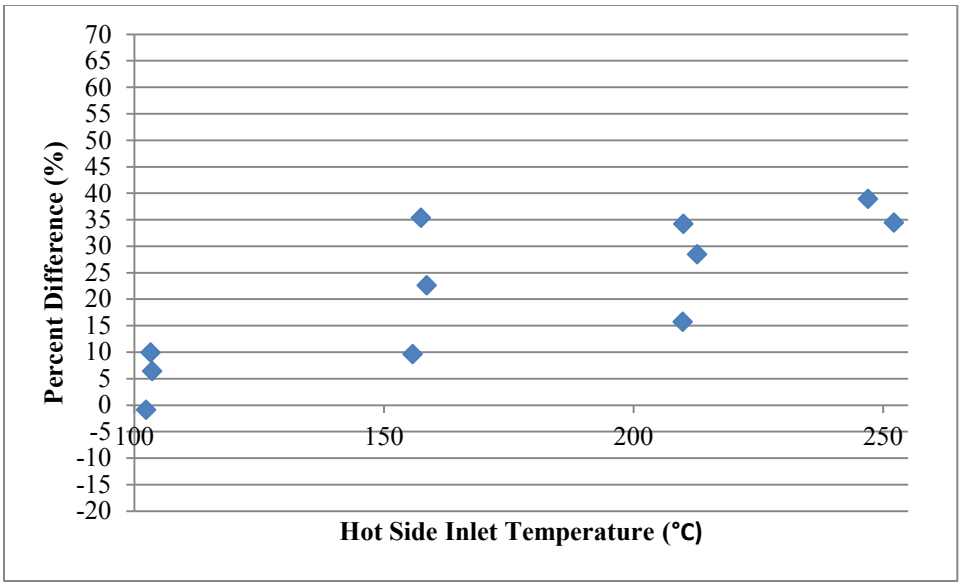


Figure 7.10: Measured power and modeled power with 5 times environmental losses percent difference.

7.3.3 Bypass Leakage

A major concern is the accuracy of the value of bypass leakage. q_{ins} represents the value that is considered bypass leakage as this is defined as the heat energy that does not flow through the thermoelectric modules but bypasses them. It is possible that the manufacturer reported conductive coefficient is not precisely as reported. It is also possible that heat leaked through small gaps in the insulation where it is butted up against the modules. In addition, the plates that hold the hot and cold side of the heat exchanger may conduct some heat resulting in a thermal short. The mounting hardware couples the plates and has a considerable effect on the small test system. The model assumes bypass leakage as the dead space between the hot side and cold side of the heat exchanger. It does not consider other potential bypass paths. Overall, this assumption provides another means of displaced energy that is not accounted for by the model in the base case. An energy balance of the hot side and cold side experimental data shows that the bypass is approximately 4.25 times the value provided for the insulation in the base case. Results are shown in Figure 7.11 and Figure 7.12 with significant improvements in prediction. The 100°C tests are excluded from the remainder of the reported data due to the limited ability to see any worthwhile value changes as values tend to be overcompensated in the lower energy cases. The amount of power output at the low temperature difference across the module is too small and results in drastic changes when input parameter changes occur.

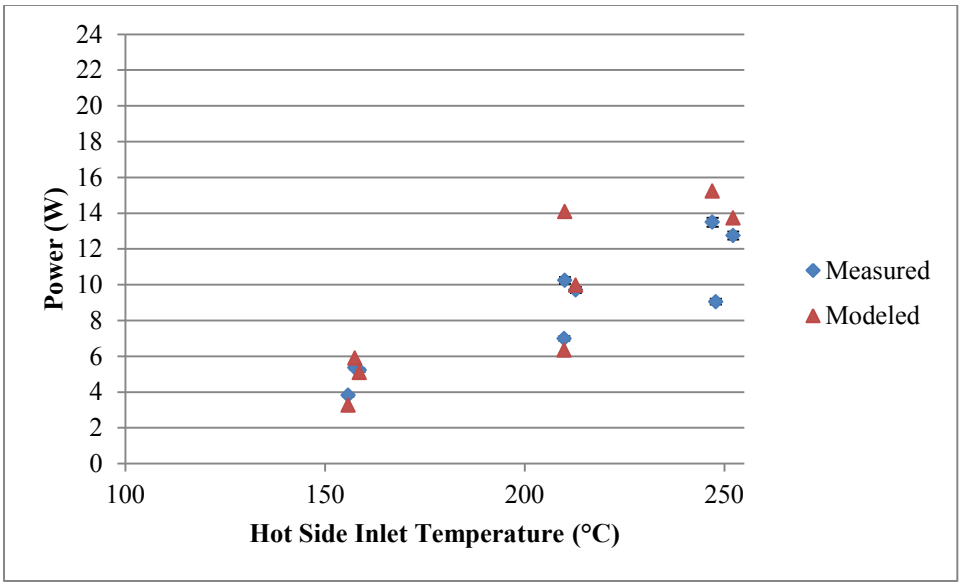


Figure 7.11: Measured power and modeled power with 4.25 times bypass leakage results.

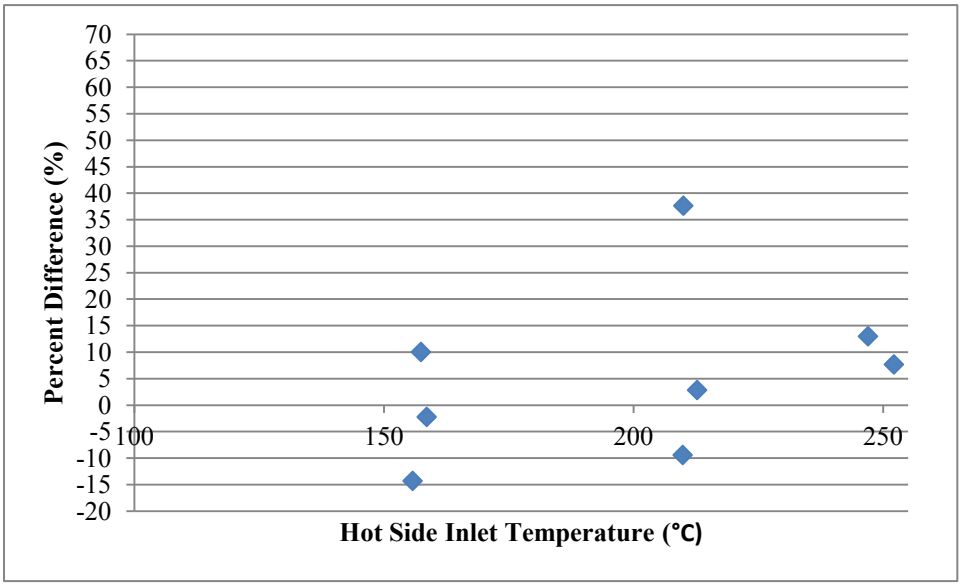


Figure 7.12: Measured power and modeled power with 4.25 times bypass leakage percent difference.

In Figure 7.12, one data point remains at approximately 40 percent difference. This point is caused by a corrupt data region in *FluidProp*. *FluidProp* values are determined from continuous functions in which the data changes as expected. At random points throughout the data, the *FluidProp* values will start to drop off to a negative value and eventually an undefined point. Probing the region showed that the inputs were unsolvable when bypass leakage was set slightly less than 4.25 times the original bypass leakage for this data point. When the input was set to be

slightly more than 4.25 times the original value, the percent difference decreased significantly. This would appear to be within reason based on the remainder of the data points. Figure 7.14 shows that the point moves back into an expected result based on surrounding test data. This is because the slight manipulation of inputs causes return to correct and continuous *FluidProp* values.

Bypass leakage is combined with the environmental losses to provide more accuracy based on the analysis of the experimental data as seen in Figure 7.13 and Figure 7.14. Two of the key contributors are compensated for based on apparent energy transfer seen in the data. The improvement in results provides for promising analysis through simulation. Additional potential causes of discrepancies in the measured versus modeled results are explored in the subsequent sections.

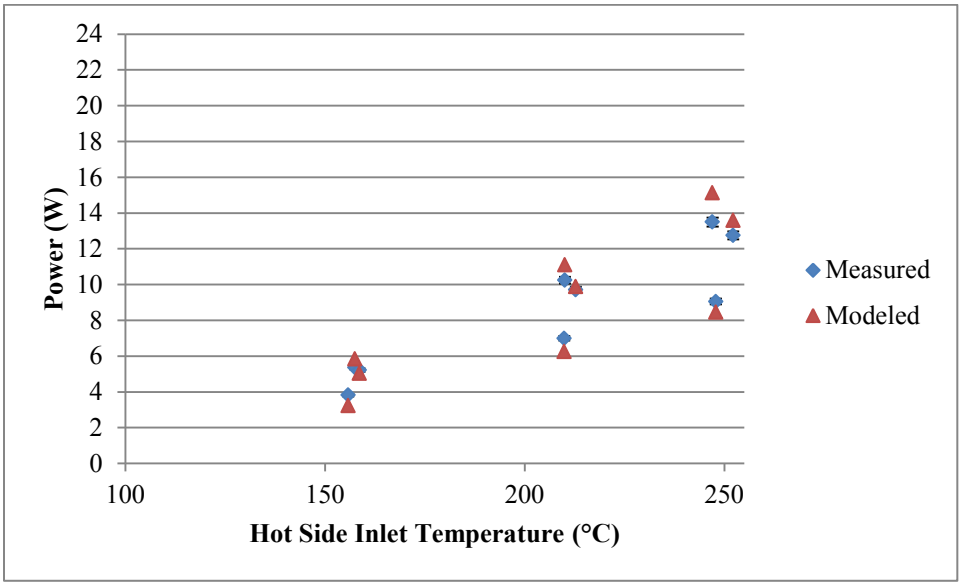


Figure 7.13: Measured power and modeled power with 5 times environmental losses and 4.25 times bypass leakage results.

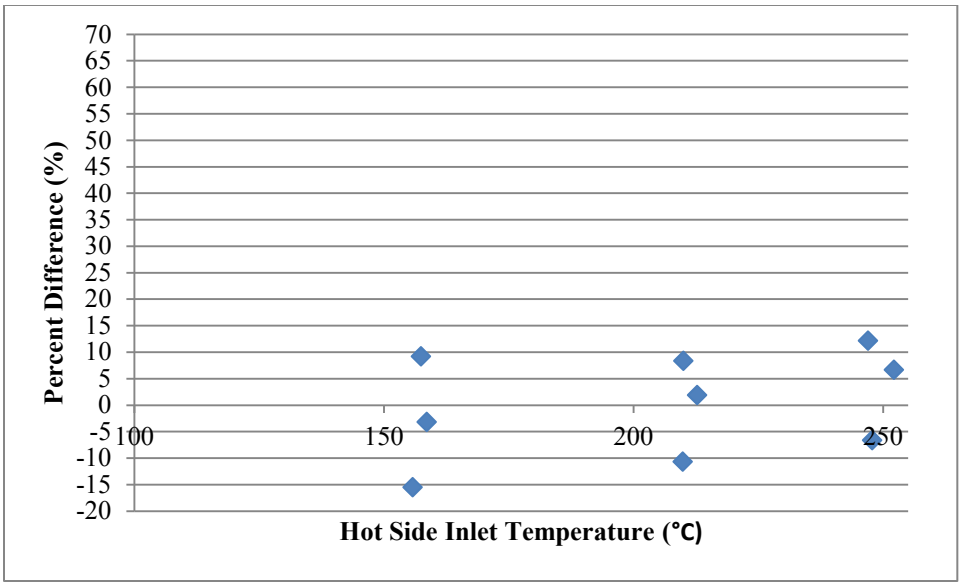


Figure 7.14: Measured power and modeled power with 5 times environmental losses and 4.25 times bypass leakage percent difference.

7.3.4 Convective Coefficient Uncertainty Range

It is known that the convective correlations can vary by +/- 20%. To better understand the impact of the uncertainty in the convective coefficient on model predictions the convection coefficient was increased and decreased by 20% and simulations were performed at these high and low values. Figure 7.15 demonstrates the acceptable range of data based on convective coefficient uncertainty for each of the 100 CFM flow rate tests. Figure 7.16 shows the uncertainty range for the 75 CFM flow rate tests. Figure 7.17 exhibits the acceptable range for the convective coefficient for the 50 CFM flow rate tests. For all tests shown, the measured data falls well within the simulation range and is acceptable. The uncertainty in the convective coefficient has a significant impact on simulated performance and must be acknowledged in any future simulation/optimization work using the thermoelectric component model.

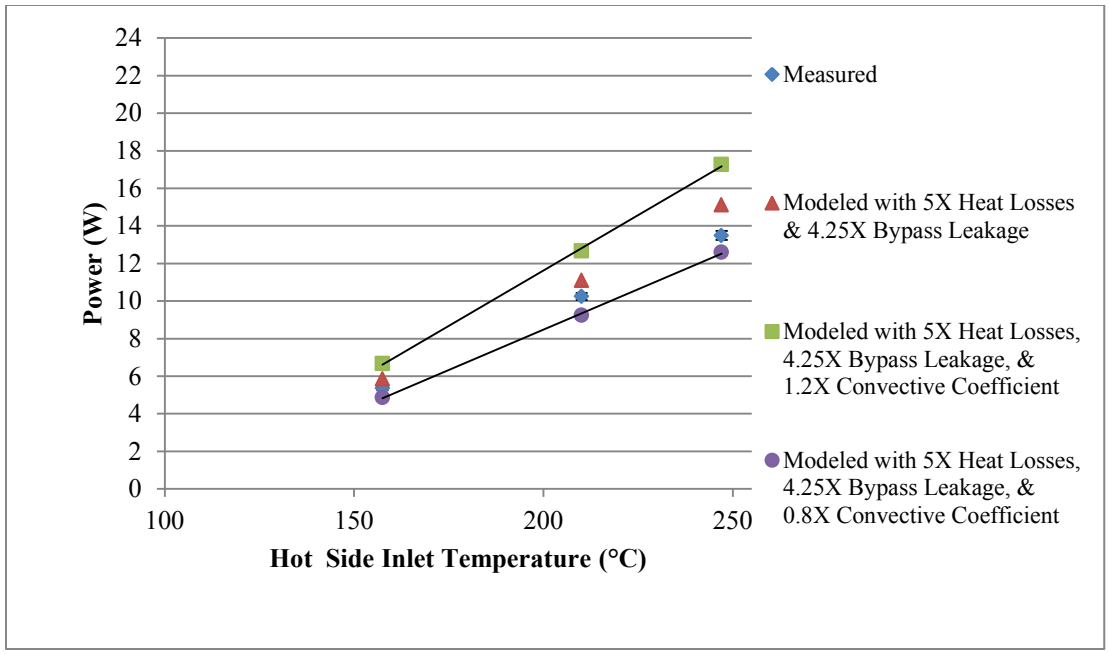


Figure 7.15: Convective coefficient uncertainty range for 100 CFM flow tests.

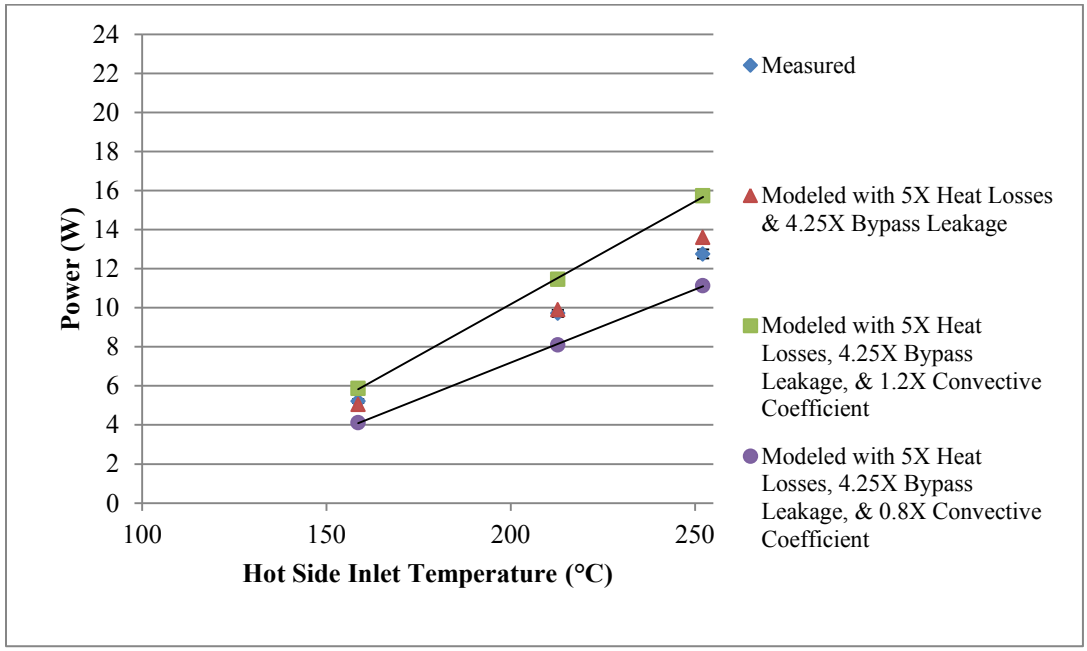


Figure 7.16: Convective coefficient uncertainty range for 75 CFM flow tests.

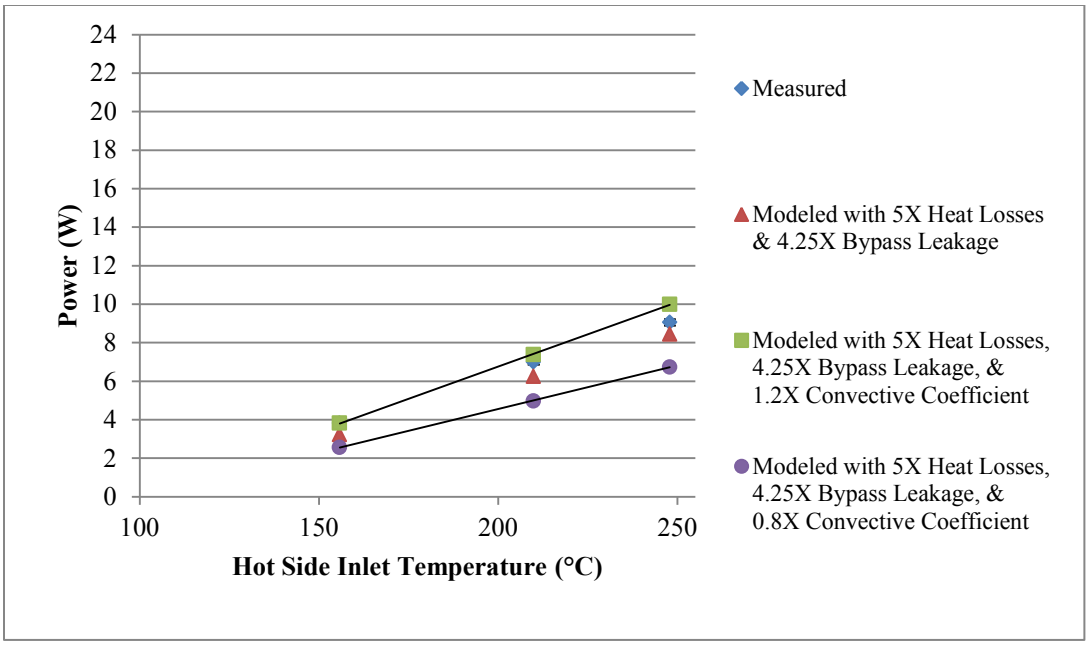


Figure 7.17: Convective coefficient uncertainty range for 50 CFM flow tests.

7.3.5 Thermal Contact Resistance Uncertainty

Another input parameter that could be an issue is the thermal contact resistance. A layer of *OMEGATHERM* paste is applied to both sides of each of the modules. Typically contact resistance is extremely small and quite difficult to measure except for specifically designed experimental setups. Similar to the heat spreading tests, with the capabilities of the lab equipment, it is impossible to provide a definitive value for contact resistance. The average contact resistance determined through experimentation in section 4.4.3 is used for the simulation analysis. This is the best reasonable assumption that can be made with regards to this parameter. This contact resistance was determined under 200°C and 100 CFM operating conditions.

Also, thermal paste is believed to vary with temperature but there is insufficient data to determine a proper scaling factor. Between the uncertainty in the conductivity at the various test set point temperatures and the assumed average value as determined in the heat spreading testing, there is considerable overall uncertainty in the value used. Thermal expansion of the paste may cause a slight shifting of thermoelectric modules which leads to additional uncertainty. Therefore, a ranged simulation is provided to understand the potential impact of contact resistance uncertainty. A thermal resistance of 50% and 300% of the measured contact resistance was used in simulations. Figure 7.18 shows the mentioned range of variation in the contact resistance input parameter for the 100 CFM cases. Figure 7.19 shows the same range but for the 75 CFM cases and Figure 7.20 shows the uncertainty range for the 50 CFM cases. For the 50 CFM case, the measured data actually falls outside of the estimated allowable range. This is likely because the modeled results with 5 times heat losses and 4.25 times bypass leakage over

predict for the lower flow rate cases. The data ranges shown in this section are purely speculation as to reason for some discrepancy with little supporting analysis to provide substantial proof. Modeled results with bypass leakage, environmental heat losses, and convective coefficient compensated for can accurately predict thermoelectric power unit performance as demonstrated in section 7.3.4.

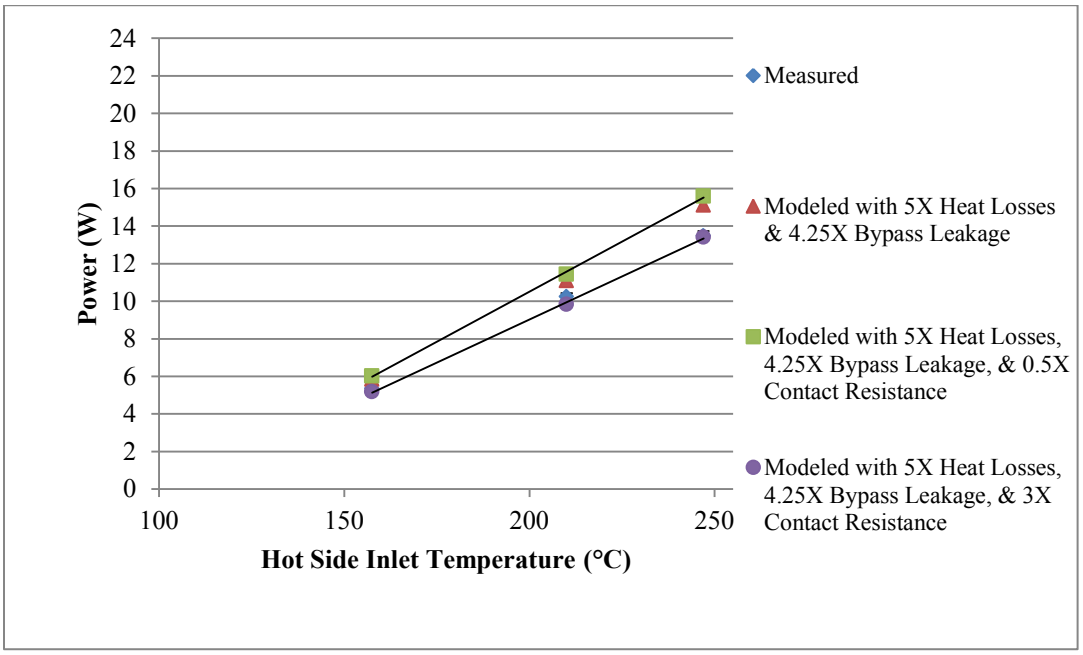


Figure 7.18: Thermal contact resistance range for 100 CFM flow tests.

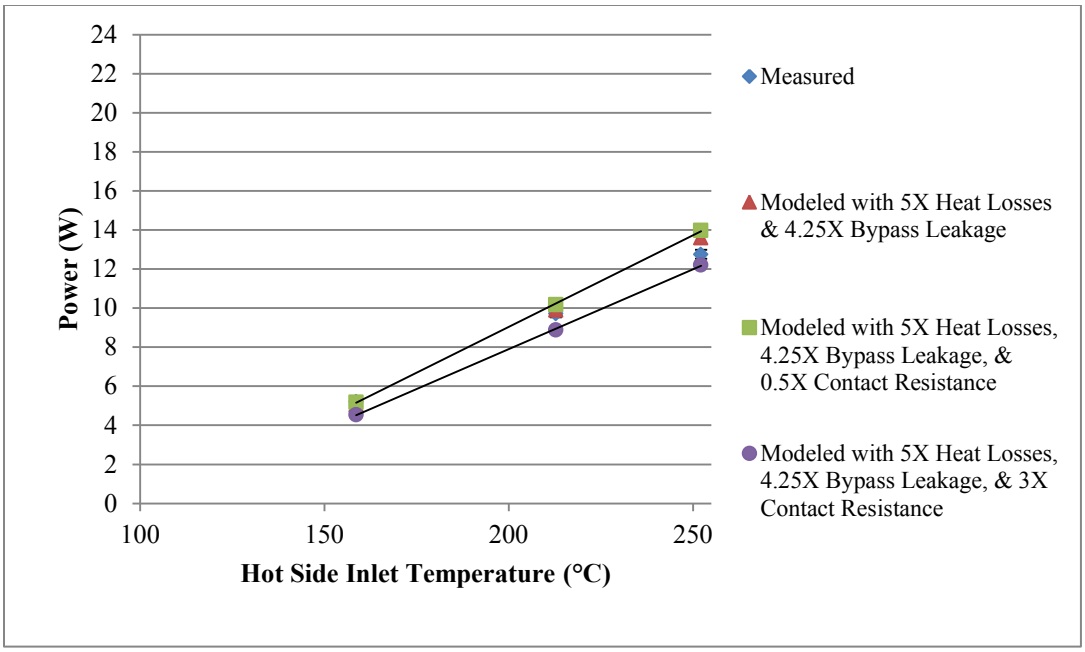


Figure 7.19: Thermal contact resistance range for 75 CFM flow tests.

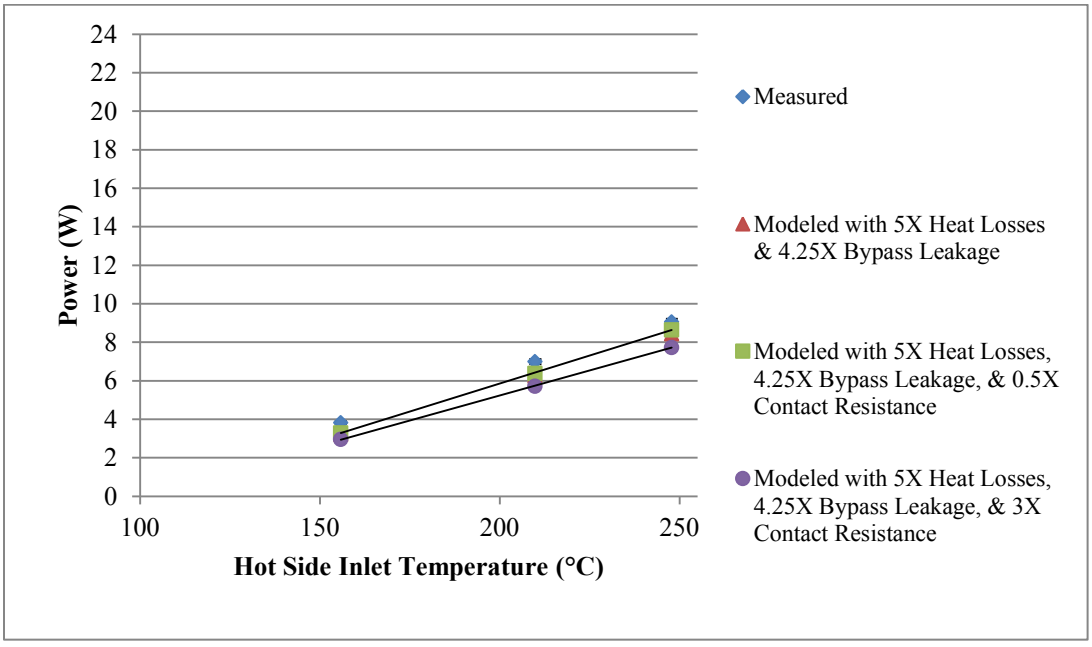


Figure 7.20: Thermal contact resistance range for 50 CFM flow tests.

7.4 Summary

Base case data is reported and several discrepancies exist between the measured and modeled power recovery results. Several potential issues are identified and each is explored through analysis of experimental data. Variations are made to the simulation input parameters and new results are reported. Based on literature, it is clear that the properties are temperature dependent, especially the thermoelectric module internal electrical resistance. Adjustments are made to compensate for the temperature dependence. It is apparent through the data analysis and adjusted parameters that bypass leakage can be critical. Environmental heat losses also contribute to the difference between the measured and modeled power results but with only minor effects. The uncertainty in the convective coefficient model provides an ample range in which the experimental results adequately match the simulated results. Thermal contact resistance is likely a contributor to the difference in measured and modeled power but limited data allows for speculation only. Experimental improvements are required to limit the effects of each of the proposed reasons for data discrepancies. Overall, the model accurately predicts the system performance when known issues are reasonably compensated for.

8 Concluding Remarks

8.1 Summary of Results

As stated in Chapter 1, the overall goal of this thesis was to improve, validate, and implement a thermoelectric subsystem model in Thermoelectric Power System Simulator (TEPSS). To achieve this goal, module mismatch was quantified through modeling and validated through experiments. Thermal resistance that results from heat spreading in three dimensions is analytically modeled, simulated numerically for validation, and experimentally verified. A robust thermoelectric power unit model was developed, programmed into a component for use in TEPSS, and a preliminary experimental validation was conducted.

Understanding module mismatch which is the discrepancies in predicted power to actual power recovered when linking thermoelectric modules electrically in series or parallel was of importance. It was believed that modules produced less power than predicted by simple modeling means so improved models were developed. Experimental data was gathered by comparing thermoelectric modules with varying Seebeck coefficients and internal electrical resistances to confirm the module mismatch models. Each module has these specific parameters that affect their voltage-current relationship, thus affecting their maximum power produced. Overall, mismatch was found to be an insignificant power prediction issue when coupling two modules, even when the values of the parameters differed greatly. Parameter ratios that strayed from unity caused thermoelectric modules to not perform as expected through simple superposition of individual modules' current-voltage relationship but the effect is small, less than 10% in extreme mismatch cases. Power ratios defined in equations (3.2.12) and (3.3.16) provide accurate prediction for module mismatch, even mismatch that is insignificant for modules that have close key parameters.

Heat spreading in three dimensions is currently not considered in any known system models for thermoelectric power units from literature. Accounting for this phenomenon provides a better understanding of differences between thermoelectric power unit model predictions and experimental results. An analytical approach was developed to compensate for this phenomenon as opposed to an experimental or numerical approach because a quick and accurate solution is desired. The obtained value is referenced in the thermoelectric power unit model and can be used by the simulation and optimization routines in TEPSS. The use of an existing model found in the literature with the described modifications provides an effective thermal resistance. The analytical model was compared to numerical simulations with good agreement under normal scenarios. Experimental data was gathered, comparing the results to the analytical solution for verification. Overall, results were promising, most within a factor of 2 when compensating for the thermal contact resistance. It was shown that the better prediction of thermal contact resistance strengthened the observation of the spreading resistance. Acceptable results were obtained with regards to the existence of spreading resistance as it could be seen through the

experimental data gathered that spreading resistance could be 20-30% of the total thermal resistance for cases where the heat source was much smaller than the heat sink. The accuracy of the analytical method for finding the spreading resistance was promising. However, a better method for quantifying contact resistance is needed to fully validate the accuracy of the heat spreading model and provide additional knowledge for further improvements to the model.

The developed thermoelectric power unit system model serves a great importance to TEPSS. It is significant for use throughout the industry for prediction of power based on the many available inputs into the system. The system model which accounts for several options for inputting TEG parameters, heat exchanger type selection, fin type selection, size constraints and assembly selection, and has three dimensional heat spreading factored in, allows for a sound prediction of power. The preliminary experimental validation was conducted with the model and experimental overall performance results, in general, within ~15%. This was well within the model input and experimental uncertainties. Key modeling uncertainties included thermal contact resistance, convective coefficient correlation function, bypass leakage, and losses to the environment. The basic principle behind the system model is the balance of heat energy throughout the system taking into account the thermoelectric phenomena of heat conversion to electrical energy. Varying operating conditions are available to be analyzed by the model. Several important factors that feed into the system model are capable of being calculated or directly provided. There are five different means of determining and entering thermoelectric module key performance parameters used by the power unit model. These include, direct input to the system, calculation as a result of P-type and N-type leg design inputs, two methods for calculating the key parameters based on thermoelectric cooler manufacturer datasheets, and a method for calculating the key parameters from the thermoelectric generator manufacturer datasheets. Three types of fins, one type having two configuration options, are implemented in the model. Rectangular straight fins, offset strip fins, and aligned or staggered pin fins are built and performance is calculated using sound industry accepted methods. Three dimensional heat spreading developed in Chapter 4 was also utilized in the calculations of overall fin system performance. A different fin type can be chosen for each side of the thermoelectric heat exchanger. The TEPSS thermoelectric model can be used as an individual component model or as part of a more complex system for both steady state simulations and optimization studies, where a multitude of design variables can be adjusted.

The model for the thermoelectric power unit has been developed into a software component in the Thermoelectric Power System Simulator. This combination of the TEPSS computer software provides insight into the feasibility of implementing a thermoelectric power unit heat exchanger into an actual heat recovery platform. The *tepowerunit* can be simulated individually to determine the capabilities of a proposed or existing device. Various parameters of the *tepowerunit* can be adjusted to optimize an overall system, which may be reducing overall cost per energy generated or payback time. With the TEPSS software architecture, it is possible to expand or manipulate *tepowerunit* to incorporate newly developed thermoelectric heat exchanger

models. This allows for a performance comparison across similar models throughout the research community. It also allows for a simple means of comparison to experimental data.

A thermoelectric power unit device was built for the purpose of testing the power unit model. Base case data was reported and several discrepancies existed between the measured and modeled power recovery initial results. Several potential issues were identified and each was explored through analysis of experimental data. Variations are made to the simulation input parameters and new results were reported. Based on literature, it is clear that the thermoelectric properties are temperature dependent as originally assumed, especially the thermoelectric module internal electrical resistance. Adjustments were made to compensate for the temperature dependence. It is apparent through the data analysis and adjusted parameters that the initial simulation bypass leakage required a revisit. Environmental heat losses also contributed to the difference between the measured and modeled power results but with only minor effects. The overall performance for the cases tested was highly sensitive to convective coefficient correlations. When accounting for the potential uncertainty in the convective correlations experimental results were within the models uncertainty range. Thermal contact resistance is likely a contributor to the difference in measured and modeled power but limited data allows for speculation only. Experimental improvements are required to limit the effects of each of the proposed reasons for data discrepancies. Overall, the model accurately predicted the system performance when known issues are reasonably compensated for.

8.2 Contributions to the Thermoelectric Field

Several contributions are made to the thermoelectric research community. First and foremost, the thermoelectric power unit model has been advanced beyond any general model currently presented in the literature. The general thermoelectric power unit model permits simulation of a wealth of concepts and with a wide range of design variable adjustments. This leads to a customizable device model which is essential for optimization. The advanced model was programmed into the *tepowerunit* component in the Thermoelectric Power System Simulator. The open source code can be edited to adapt to further advancements or new modeling desires. The use of the complete TEPSS package would allow one to consider the creation of an actual thermoelectric power unit device for use in industrial applications or automobiles.

Additionally, the advanced thermoelectric power unit model considers three dimensional heat spreading resistance. This resistance should be utilized by other models in the research community to enhance performance predictions of those models. The heat spreading model can easily be applied to existing thermoelectric power unit models through the analytical equations described in Chapter 4.

Furthermore, module mismatch was shown theoretically and experimentally to be relatively insignificant when using similar thermoelectric modules. As this phenomenon has not been

addressed throughout the thermoelectric research field, this result is a significant contributor. The developed and verified model is a unique piece of work delivered from this thesis.

8.3 Improvements and Future Work

Certain improvements and future work are recommended to resolve questionable issues that arose from this research. Most are experiment related but a simulation issue exists. It is known that *FluidProp*, the database used to look up thermodynamic properties based on input conditions, has several regions of corrupt data as determined in Section 7.3. A method for resolving these bad regions other than avoiding them needs to be developed. This will allow for completion of all simulations performed in comparison to the experiment data for the thermoelectric power unit as provided in Chapter 7. Error in *FluidProp* could also lead to erroneous optimization points when using TEPSS.

The experimental setup used for the heat spreading tests and the thermoelectric power unit tests has several areas for improvement. Although many aspects of the test bed are useful and a myriad of data can be taken, additional data points are required. Boundary condition temperature readings are taken at the manifolds. There would be a benefit to obtain temperature readings at the exact ends of the power unit itself. Although only inches away, this would remove some issues with heat losses or discredit them as issues altogether. Several additional upstream and downstream temperature measurements could be taken to determine more characteristics of the test bed piping system and flow temperature losses. In addition to temperature measurements, it is far more critical to obtain more flow rate data at several points including the inlets and outlets of each side of the power unit. This would allow the user to better characterize the flow losses through the system and provide this data to the power unit model and TEPSS simulation. Resolution of the losses of mass flow from the system can provide more precise solutions.

Extensive research with regards to thermal contact resistance is required. The paste used for the heat spreading tests and thermoelectric power unit tests was unpredictable with regards to performance. The thickness of paste application varied from test to test and thermal expansion may have changed the contact resistance during tests at different operating temperatures. Characterization of the paste used will allow more precise values or possibly a model to be implemented into the thermoelectric power unit model. Other thermal interface materials could be explored as options for use with the existing test device. Reducing or removing the uncertainty associated with the thermal contact resistance is of great importance and the next step for a realistic power unit development.

The thermoelectric power unit model has a couple areas for improvement. As determined through the experiment, the model wasn't accurately compensating for bypass leakage and heat losses. Bypass leakage was approximately 4.25 times different for each of the test cases. Heat losses ranged from 2 times to 9 times different based on operating temperatures and flow rates

meaning that the heat loss portion of the model requires improvement. Accurate modeling of each of these items is required for precision in the results provided by the thermoelectric power unit model.

The thermoelectric power unit model is highly customizable but the experimentation was limited to laboratory resources. More comprehensive testing to verify the accuracy of the model is required. Different fin types, including offset strip fins and aligned and staggered pin fins, with varying dimensions as well as different thermoelectric modules should be explored through experimentation in comparison with the model. This would serve as verification of some of the customizable capabilities of the model that have not been tested.

In the near future, after model improvements are made and small scale experimentation has proven the thermoelectric power unit model to be thoroughly advanced, a scaled up device becomes a worthwhile exploration. Simulations for a larger unit utilizing a greater number of thermoelectric generators can be performed and should be when minor model improvements are made. The larger power unit could be a prototype of a device to be used in modern day power plants replacing older heat exchangers and allowing useful power recovery. A large scale prototype is the next milestone in implementing thermoelectric generators for the recovery of waste heat in industry and reducing mankind’s carbon footprint.

References

[1] D. M. Rowe, “General Principles and Basic Considerations.” *Thermoelectrics Handbook: Macro to Nano.* Boca Raton, Florida: CRC Press Taylor & Francis Group, 1-1 – 1-13, 2006.

[2] A.F. Ioffe, *Semiconductor Thermoelements and Thermoelectric Cooling*. Publishing House USSR Academy of Sciences, Moscow, 1956.

[3] A. Majumdar, “Thermoelectricity in Semiconductor Nanostructures,” *Science*, vol. 33, pp. 777-778, Feb. 2004.

[4] S. Lineykin and S. Ben-Yaakov, "Modeling and Analysis of Thermoelectric Modules," *IEEE Ttansaction on Industry Applications,* vol. 43, no. 2, pp. 8, March/April 2007.

[5] E. Sandoz-Rosado and R. J. Stevens, “Experimental Characterization of Thermoelectric Modules and Comparison with Theoretical Models for Power Generation,” *Journal of Electronic Materials*, vol. 38, no. 7, pp. 1239-1244, July 2009.

[6] M. Hodes, "On one-dimensional analysis of thermoelectric modules (TEMs)," *IEEE Transactions on Components and Packaging Technologies,* vol. 28, pp. 218-229, 2005.

[7] A. Bethancourt, R. Echigo, and H. Yoshida, “Thermoelectric Conversion Analysis in a Counter-flow Heat Exchanger,” *AIP Conference Proceedings***,** 316, pp. 299-304, 1995.

[8] D. T. Crane and G. S. Jackson, “Optimization of Cross Flow Heat Exchangers for Thermoelectric Waste Heat Recovery,” *Energy Conversion and Management*, vol. 45, no. 9, pp. 1565-82, 2004.

[9] K. Smith, “An Investigation into the Viability of Heat Sources for Thermoelectric Power Generation Systems,” M.S. Thesis, Department of Mechanical Engineering, Rochester Institute of Technology, Rochester, NY, USA, 2009.

[10] J. LaManna, D. Ortiz, M. Livelli, S. Haas, C. Chikwem, B. Ray, R. Stevens, “Feasibility of Thermoelectric Waste Heat Recovery in Large Scale Systems,” *Proceedings of ASME International Mechanical Engineering Congress and Exposition*, 2008.

[11] Lawrence Livermore National Laboratory and United States Department of Energy, “Annual Energy Review 2009,” DOE/EIA-0384(2009), August 2010.

[12] S. B. Riffat and X. Ma, "Thermoelectrics: a review of present and potential applications," *Applied Thermal Engineering,* vol. 23, pp. 913-35, June 2003.

[13] R. J. Stevens "NYSERDA PON1190" unpublished, 2008.

[14] G. N. Ellison, "Maximum Thermal Spreading Resistance for Rectangular Sources and Plates With Nonunity Aspect Ratios," *IEEE Transactions on Components and Packaging Technologies,* vol. 26, no. 2, pp. 439-454, June 2003.

[15] E. J. Sandoz-Roszado, "Investigation and development of advanced models of thermoelectric generators for power generation applications". M.S. Thesis, Department of Mechanical Engineering, Rochester Institute of Technology, Rochester, NY, USA, 2009.

[16] E. E. Antonova and D. C. Looman, "Finite elements for thermoelectric device analysis in ANSYS," *24th International Conference on Thermoelectrics (ICT)* (IEEE Cat. No.05TH8854C), (Clemson, SC, USA), pp. 215–18, IEEE, 2005.

[17] M. S. Bohn, "Heat-Exchanger Effectiveness in Thermoelectric Power Generation," *Transactions of the ASME. Journal of Heat Transfer*, vol. 103, n. 4, pp. 693-698, 1981.

[18] J. Esarte, G. Min, and D. M. Rowe, "Modelling heat exchangers for thermoelectric generators," *Journal of Power Sources*, vol. 93, pp. 72-76, 2001.

[19] T. J. Hendricks and J. A. Lustbader, "Advanced Thermoelectric Power System Investigations for Light-Duty and Heavy Duty Applications: Part 1," *21st International Conference on Thermoelectrics*, pp. 381-386, 2002.

[20] M. A. Karri, "Modeling of an Automotive Exhaust Thermoelectric Generator," M. S. Thesis, Department of Mechanical and Aeronautical Engineering, Clarkson University, Potsdam, NY, USA.

[21] M. A. Karri, E. F. Thacher, and B. T. Helenbrook, "Exhaust Energy Conversion by Thermoelectric Generator: Two Case Studies," *Energy Conversion and Management*, vol. 52, n. 3, pp. 1596-1611, March 2011.

[22] E. F. Thacher and B. T. Helenbrook, "AETEG Performance Studies Report - Thermoelectric Power from Vehicle Exhaust: Phase 2B," New York State Energy Research and Development Authority, Clarkson University, Potsdam, NY, USA.

[23] R. O. Suzuki and D. Tanaka, “Mathematical simulation of thermoelectric power generation with the multi-panels,” *Journal of Power Sources*, vol. 122, pp. 201-209, 2003.

[24] R. O. Suzuki and D. Tanaka, “Mathematic simulation on thermoelectric power generation with cylindrical multi-tubes,” *Journal of Power Sources*, vol. 124, pp. 293-298, 2003.

[25] R. O. Suzuki and D. Tanaka, “Mathematic Simulation on Power Generation by Roll Cake Type of Thermoelectric Cylinders,” *25th International Conference on Thermoelectrics*, pp. 6 , 2006.

[26] D. T. Crane and G. S. Jackson, “Systems-Level Optimization of Low Temperature Thermoelectric Waste Heat Recovery,” *Proceedings of the Intersociety Energy Conversion Engineering Conference*, pp. 583-591, 2002.

[27] J. Yu and H. Zhao, “A numerical model for thermoelectric generator with the parallel-plate heat exchanger,” *Journal of Power Sources*, vol. 172, no. 1, 2007.

[28] X. Niu, J. Yu, and S. Wang “Experimental study on low-temperature waste heat thermoelectric generator,” *Journal of Power Sources*, vol. 188, no. 2, pp. 621-626, 2009.

[29] J. Rhee and A. D. Bhatt, “Spatial and Temporal Resolution of Conjugate Conduction-Convection Thermal Resistance,” *IEEE Transactions on Components and Packaging Technologies*, vol. 30, no. 4, pp. 673-682, December 2007.

[30] H. Nagayoshi, T. Kajikawa, and T. Sugiyama, “Comparison of maximum power point control methods for thermoelectric power generator,” *21st International Conference on Thermoelectrics*, pp. 450-453, 2002.

[31] H. Nagayoshi, K. Tokumisu, and T. Kajikawa, “Evaluation of multi MPPT thermoelectric generator system,” *26th International Conference on Thermoelectrics*, pp. 318-321, 2007.

[32] H. Nagayoshi, T. Nakabayashi, H. Maiwa, and T. Kajikawa, “Development of 100-W High Efficiency MPPT Power Conditioner and Evaluation of TEG System with Battery Load,” *Journal of Electronic Materials*, vol. 40, no. 5, pp. 657-661, 2011.

[33] A. Chouder and S. Silvestre, “Analysis model of mismatch power losses in PV systems,” *Journal of Solar Energy Engineering*, vol. 131, no. 2, pp.024504-1 – 024504-5, 2009.

[34] D. Picault, B. Raison, S. Bacha, J. de la Casa, and J. Aguilera, "Forecasting photovoltaic array power production subjest to mismatch losses," *Solar Energy*, vol. 84, no. 7, pp. 1301-1309, 2010.

[35] Laird Technologies, ThermaTEC™ Series - High Temperature Thermoelectric Modules, *HT8,12,F2,4040,TA*, Available: http://www.lairdtech.com/

[36] Xiamen Taihuaxing Trading Co., Ltd., Taihuaxing Thermoelectric Power Modules, *TEP1-1264-1.5*, Available: http://www.sitechina.com/thermoelectric/

[37] F. P. Incropera, D. P. DeWitt, T. L. Bergman, and A. S. Lavine, *Fundamentals of Heat and Mass Transfer*, 6th ed. Hoboken, NJ: John Wiley & Sons, Inc., 2007.

[38] M&M Metals, Aluminum Heat Sink Extrusion Profiles Flat Backs 1" to 2", *MM10551*, Available: http://mmmetals.com/

[39] Omega Engineering, Inc., OMEGATHERM® High Temperature and High Thermally Conductive Paste, *OT-201*, Available: http://www.omega.com/

[40] Z. Luo, "A Simple Method to Estimate the Physical Characteristics of a Thermoelectric Cooler from Vendor Datasheets," *Electronics Cooling*, 2008, Available: http://www.electronics-cooling.com/

[41] W. M. Kays and M. E. Crawford, *Convection Heat and Mass Transfer*, 3rd ed. McGraw-Hill, New York, 1993.

[42] E. N. Sieder and G. E. Tate, "Heat Transfer and Pressure Drop of Liquids in Tubes," *Ind. Eng. Chem.*, vol. 28, 1429-1436, 1936.

[43] B. S. Petukhov, "Heat transfer and friction in turbulent pipe flow with variable physical properties," in T. F. Irvine and J. P. Hartnett, (Editors)., *Advances in Heat Transfer*, vol. 6, Academic Press, New York, 1970.

[44] V. Gnielinski, "New Equations for Heat and Mass Transfer in Turbulent Pipe and Channel Flow," *International Chemical Engineering*, vol. 16, pp. 359-368, 1976.

[45] R. M. Manglik and A. E. Bergles, "Heat Transfer and Pressure Drop Correlations for the Rectangular Offset Strip Fin Compact Heat Exchanger," *Experimental Thermal and Fluid Sciences*, vol. 10, no. 2, pp. 171-180, 1995.

[46] H. M. Joshi and R. L. Webb, "Heat Transfer and Friction in the Offset Strip-Fin Heat Exchanger," *International Journal of Heat and Mass Transfer*, vol. 28, pp. 69-84, 1987.

[47] W. M. Kays and A. L. London, *Compact Heat Exchangers*, 3rd ed., McGraw-Hill, New York, 1984.

[48] S. Y. Kim and R. L. Webb, "Analysis of Convective Thermal Resistance in Ducted Fan-Heat Sinks," *IEEE Transactions on Components and Packaging Technologies*, vol. 29, no. 3, pp. 439-448, 2006.

[49] A. Zukauskas, "Heat Transfer from Tubes in Cross Flow," in J. P. Hartnett and T. F. Irvine, Jr. (Editors), *Advances in Heat Transfer*, vol. 8, Academic Press, New York, 1972.

[50] S. W. Churchill and M. Bernstein, "A Correlating Equation for Forced Convection from Gases and Liquids to a Circular Cylinder in Cross Flow," *Journal of Heat Transfer*, vol. 99, pp. 300-306, 1977.

[51] E. S. Gaddis and V. Gnielinski, "Pressure drop in cross flow across tube bundles," *International Chemical Engineering*, vol. 25, pp. 1-15, 1985.

[52] J. J. Kreuder, "Development of a Platform for Simulating and Optimizing Thermoelectric Energy Systems," M. S. Thesis, Department of Mechanical Engineering, Rochester Institute of Technology, Rochester, NY, USA, 2010.

[53] P. Colonna, T.P. van der Stelt, 2004, *FluidProp: a program for the estimation of thermo physical properties of fluids*, Energy Technology Section, Delft University of Technology, The Netherlands, Available: http://www.FluidProp.com

[54] H. Martin, "The generalized Lévêque equation and its practical use for the prediction of heat and mass transfer rates from pressure drop," *Chemical Engineering Science*, vol. 57, pp. 3217-3223, 2002.

Appendix A

```
classdef tepowerunit < handle

    %TEPOWERUNIT - Thermoelectric Power Unit
    %The thermoelectric power unit is a heat exchanger that utilizes
    %thermoelectric generators to recover waste heat from various
    %applications. The power unit can be set up with a hot flow fluid path
    %and cold fluid flow path or with a constant heat or isothermal cool
    %side. Five options exist for determining thermoelectric key
    %parameters. They include inputting predetermined parameters through
    %empirical methods, two methods for calculating parameters from
    %thermoelectric cooler (TEC) data sheets, calculating parameters from
    %thermeoelctric generator (TEG) data sheets, and determing the
    %parameters from material properties and geometry inputs. Each side of
    %the heat exchanger has several options for fin type. These options
    %include rectangular straight fins, offset strip fins, aligned pin
    %fins, and staggered pin fins. There is a parameter check function
    %which ensures that realistic fin parameters are input into the power
    %unit model to keep from having solution errors in TEPSS. The power
    %unit is solved through discretization of the device and solving each
    %zone. The performance of the zones are combined and overall
    %performance is determined. When the residual equations are solved to
    %have residuals within the tolerance set by the user, the tepowerunit
    %object solution is converged.

    properties (SetAccess=private)
        module %Properties related to the thermeoelctric modules (TEM's)
        fins %Properties related to the heat exchanger fins
        unit %Properties related to the heat exchanger unit
        options %{method for finding TEM parameters,type of heat exchanger,
                %type of hot side fins, type of cold side fins}
        e %Error functions array
        power %Total power recovered by the thermoelectric power unit
        q_h %Total heat rate entering the hot side of the TEM's
        q_c %Total heat rate exiting the cold side of the TEM's
        q_c_tot %For isothermal and constant heat rate HX
        q_loss_h %Total heat rate lost from the hot side of the HX
        q_loss_c %Total heat rate lost from the cold side of the HX
        q_ins %Heat rate through the insulation
        specheat_h %Hot side specific heat
        specheat_c %Cold side specific heat
        costinfo %Stores all cost information related to the tepowerunit
        onoff %Can turn tepowerunit unit TEPSS component on or off
        parameters %Stores all inputs to the system
        press_loss %Stores hot and cold side pressure losses for each zone
        surf_temp_hot=500; %Initial value for hot side surface temp
        surf_temp_cold=300; %Initial value for cold side surface temp
    end

    methods

        function obj = tepowerunit(parameters)
            %Constructor - Takes the user provided initial parameters and
            %passes these structures in to create the object which defines
```

```
    %the unit.

    obj.parameters=parameters; %Contains all the sub-structures
    obj.fins=parameters.fins; %Defines the HX fins
    parameters.unit.num = round(parameters.unit.num); %Defines the
                                                      %number of zones
    obj.module=parameters.module; %Defines the TE modules
    obj.unit=parameters.unit; %Defines the power unit
    obj.options=parameters.options; %Defines the options which
                                    %prompt certain calculations
    obj.costinfo = parameters.cost; %Defines cost information
    obj.module_parameters; %Runs function module_parameters(obj)
    obj.unit_setup; %Runs function unit_setup(obj)
end

function e = compute(obj, node1, node2, node3, node4, onoff)
    %The compute function provides the main computational function
    %in which other functions are called and  residuals are
    %determined to be satisfied, thus completing the simulation.

    %Sets up nodes to correspond to heat exchanger locations
    if strcmp(obj.options{2},'isothermal')||...
            strcmp(obj.options{2},'constantheatrate')
        %Set up for isothermal or constant heat rate heat exchanger
        cisothermal=node1;
        hfluidin=node3;
        hfluidout=node4;
    else
        %Set up for counter flow or parallel flow heat exchanger
        cfluidin=node1;
        cfluidout=node2;
        hfluidin=node3;
        hfluidout=node4;
    end

    %Option to turn 'tepowerunit' on or off
    obj.onoff = onoff;

    if obj.onoff == 1
        %Error functions if 'tepowerunit' is turned off
        obj.e(1) = hfluidout.mdot - hfluidin.mdot;
        obj.e(2) = cfluidout.mdot - cfluidin.mdot;
        obj.e(3) = hfluidin.temp - hfluidout.temp;
        obj.e(4) = cfluidout.temp - cfluidin.temp;
        obj.e(5) = hfluidin.press - hfluidout.press;
        obj.e(6) = cfluidin.press - cfluidout.press;
        %Error functions array
        e = obj.e;

    elseif obj.onoff == 0
        %Solution process if 'tepowerunit' is turned on
        if strcmp(obj.options{2},'isothermal')||...
                strcmp(obj.options{2},'constantheatrate')

            %Setup calculations for isothermal/constant heat rate
```

```
        %heat exchanger

        %Calculate pressure drop and thermal resistance for hot
        %side
        [obj.fins.h_resist, obj.unit.h_press,...
            obj.specheat_h]=obj.precalc(hfluidin,...
            hfluidout,'hot',obj.options{3});

        %UA value that goes into Fsolve equations
        obj.unit.UA_h=1/(obj.fins.h_resist+...
            obj.unit.therm_contact_res/obj.unit.zone_mod_area);

        %Zone temperature calculations
        htemp=zeros(1,obj.unit.num+1);
        htemp(1)=hfluidin.temp;

        %Zone mass flow
        hmdot=hfluidin.mdot;

        %Zone power and heat rates are set to zero so they can
        %be computed for each zone and then compiled for the
        %whole heat exchanger
        obj.power=0;
        obj.q_h=0;
        obj.q_c=0;
        obj.q_loss_h=0;
        obj.q_ins=0;

        %Surface temp is set to zero to reset it before the
        %zone surface temps are calculated for the next
        %iteration
        obj.surf_temp_hot=0;

        if strcmp(obj.options{2},'isothermal')
            %Isothermal conditions per zone
            ctemp=cisothermal.temp;
            obj.q_c_tot=0;
        elseif strcmp(obj.options{2},'constantheatrate')
            %Constant heat rate conditions per zone
            constant_qc=cisothermal.heatrate/obj.unit.num;
        end

    else
        %Setup calculations for counter flow/parallel flow heat
        %exchangers

        %Calculate pressure drop and thermal resistance for hot
        %side
        [obj.fins.h_resist, obj.unit.h_press,...
            obj.specheat_h] = obj.precalc(hfluidin,...
            hfluidout,'hot',obj.options{3});

        %Calculate pressure drop and thermal resistance for
        %cold side
        [obj.fins.c_resist, obj.unit.c_press,...
```

```
            obj.specheat_c] = obj.precalc(cfluidin,...
            cfluidout,'cold',obj.options{4});

        %UA values that go into Fsolve equations
        obj.unit.UA_h=1/(obj.fins.h_resist+...
            obj.unit.therm_contact_res/obj.unit.zone_mod_area);
        obj.unit.UA_c=1/(obj.fins.c_resist+...
            obj.unit.therm_contact_res/obj.unit.zone_mod_area);

        %Zone temperature calculations
        htemp=zeros(1,obj.unit.num+1);
        ctemp=zeros(1,obj.unit.num+1);
        htemp(1)=hfluidin.temp;
        ctemp(1)=cfluidout.temp;

        %Zone mass flows
        hmdot=hfluidin.mdot;
        cmdot=cfluidout.mdot;

        %Zone power and heat rates are set to zero so they can
        %be computed for each zone and then compiled for the
        %whole heat exchanger
        obj.power=0;
        obj.q_h=0;
        obj.q_c=0;
        obj.q_loss_h=0;
        obj.q_loss_c=0;
        obj.q_ins=0;

        %Surface temp is set to zero to reset it before the
        %zone surface temps are calculated for the next
        %iteration
        obj.surf_temp_hot=0;
        obj.surf_temp_cold=0;

    end

    %Initial guesses for Fsolve depending on HX type
    if strcmp(obj.options{2},'isothermal')||...
            strcmp(obj.options{2},'constantheatrate')

        if strcmp(obj.options{1},'option1')||...
                strcmp(obj.options{1},'option3')||...
                strcmp(obj.options{1},'option4')||...
                strcmp(obj.options{1},'option5')
            %Based on number of equations used in Fsolve
            %calculations - 9 equations
            y0=[1;1;1;1;1;1;1;1;1];

        elseif strcmp(obj.options{1},'option2')
            %Based on number of equations used in Fsolve
            %calculations - 11 equations
            y0=[1;1;1;1;1;1;1;1;1;1;1];
        end
```

```
else
    if strcmp(obj.options{1},'option1')||...
            strcmp(obj.options{1},'option3')||...
            strcmp(obj.options{1},'option4')||...
            strcmp(obj.options{1},'option5')
        %Based on number of equations used in Fsolve
        %calculations - 12 equations
        y0=[1;1;1;1;1;1;1;1;1;1;1;1];

    elseif strcmp(obj.options{1},'option2')
        %Based on number of equations used in Fsolve
        %calculations - 14 equations
        y0=[1;1;1;1;1;1;1;1;1;1;1;1;1;1];
    end

end

%Calculations made for each zone
for ii=1:obj.unit.num

    if strcmp(obj.options{2},'isothermal')||...
            strcmp(obj.options{2},'constantheatrate')

        if strcmp(obj.options{2},'isothermal')
            %Fsolve set up for isothermal conditions
            solve_zone_isothermal1=...
                @(x)obj.solve_zone_isothermal(x,...
                htemp(ii),ctemp,hmdot);

        elseif strcmp(obj.options{2},'constantheatrate')
            %Fsolve set up for constant heat rate
            %conditions
            solve_zone_isothermal1=...
                @(x)obj.solve_zone_isothermal(x,...
                htemp(ii),constant_qc,hmdot);

        end

        %Options for Fsolve
        options_fsolve = optimset('Display','off');

        %Runs Fsolve based on above conditions, initial
        %guesses, and options
        y=fsolve(solve_zone_isothermal1,y0,options_fsolve);

        %Hot side temp (Th) for each zone
        htemp(ii+1)=y(5);

        %Power and heat rates for each zone compiled for HX
        obj.power=obj.power+y(1);
        obj.q_h=obj.q_h+y(2);
        obj.q_c=obj.q_c+y(3);
        obj.q_loss_h=obj.q_loss_h+y(8);
        obj.q_ins=obj.q_ins+y(9);
```

```
        %Used to determine an average surface temperature
        %in calculations below
        obj.surf_temp_hot=obj.surf_temp_hot+y(5);

        if strcmp(obj.options{2},'isothermal')
            %Total cold side heat rate based on each zone
            %heat rate, used in error equations
            obj.q_c_tot=obj.q_c_tot+y(4);
        elseif strcmp(obj.options{2},'constantheatrate')
            %Constant heat rate for the entire HX
            obj.q_c_tot=constant_qc*obj.unit.num;
            %Cold side temp (Tc) for each zone
            ctemp(ii)=y(4);
        end


        %Setting all the error equations or convergence
        %criteria

        %Conservation of mass convergence
        obj.e(1)=hfluidin.mdot-hfluidout.mdot;

        %Pressure drops, used in conservation of pressure
        %convergence
        obj.press_loss.h_press(ii)=obj.unit.h_press;
        deltaP_h=sum(obj.press_loss.h_press);

        %Conservation of pressure convergence
        obj.e(2)=hfluidin.press-hfluidout.press-deltaP_h;

        %Conservation of hot side energy convergence
        obj.e(3)=hfluidin.mdot*(hfluidin.enthalpy-...
            hfluidout.enthalpy)-obj.q_h-obj.q_loss_h-...
            obj.q_ins;

        if strcmp(obj.options{2},'isothermal')
            %Conservation of cold side energy convergence
            obj.e(4)=cisothermal.heatrate-obj.q_c_tot;
        elseif strcmp(obj.options{2},'constantheatrate')
            %Temperature balance convergence
            obj.e(4)=cisothermal.temp-mean(ctemp);
        end

        %Error functions array
        e = obj.e;

    else
        %Fsolve set up for counter flow conditions
        solve_zone1=@(x)obj.solve_zone(x,htemp(ii),...
            ctemp(ii),hmdot,cmdot);

        %Options for Fsolve
        options_fsolve = optimset('Display','off');
```

```
%Runs Fsolve based on above conditions, initial
%guesses, and options
y=fsolve(solve_zone1,y0,options_fsolve);

%CHANGE WORDING %Hot side temp (Th) for each zone
htemp(ii+1)=y(6);
%CHANGE WORDING %Cold side temp (Tc) for each zone
ctemp(ii+1)=y(7);

%Power and heat rates for each zone compiled for HX
obj.power=obj.power+y(1);
obj.q_h=obj.q_h+y(2);
obj.q_c=obj.q_c+y(3);
obj.q_loss_h=obj.q_loss_h+y(10);
obj.q_loss_c=obj.q_loss_c+y(11);
obj.q_ins=obj.q_ins+y(12);

%Used to determine average surface temperatures in
%calculations below
obj.surf_temp_hot=obj.surf_temp_hot+y(4);
obj.surf_temp_cold=obj.surf_temp_cold+y(5);


%Setting all the error equations or convergence
%criteria

%Conservation of mass convergence
obj.e(1)=hfluidin.mdot-hfluidout.mdot;
obj.e(2)=cfluidin.mdot-cfluidout.mdot;

%Pressure drops, used in conservation of pressure
%convergence
obj.press_loss.h_press(ii)=obj.unit.h_press;
deltaP_h=sum(obj.press_loss.h_press);
obj.press_loss.c_press(ii)=obj.unit.c_press;
deltaP_c=sum(obj.press_loss.c_press);

%Conservation of pressure convergence
obj.e(3)=hfluidin.press-hfluidout.press-deltaP_h;
obj.e(4)=cfluidin.press-cfluidout.press-deltaP_c;

%Conservation of hot side energy convergence
obj.e(5)=hfluidin.mdot*(hfluidin.enthalpy-...
    hfluidout.enthalpy)-obj.q_h-obj.q_loss_h-...
    obj.q_ins;
%Conservation of cold side energy convergence
obj.e(6)=cfluidin.mdot*(cfluidout.enthalpy-...
    cfluidin.enthalpy)-obj.q_c+obj.q_loss_c-...
    obj.q_ins;

%Error functions array
e = obj.e;
```

```
                end

            end

            %Determines average surface temperatures
            if strcmp(obj.options{2},'isothermal')||...
                    strcmp(obj.options{2},'constantheatrate')

                obj.surf_temp_hot=obj.surf_temp_hot/obj.unit.num;

            else
                obj.surf_temp_hot=obj.surf_temp_hot/obj.unit.num;
                obj.surf_temp_cold=obj.surf_temp_cold/obj.unit.num;
            end

        else
            %Forces component to either be on or off
            disp('solver_inputs.removable must be an array containing
only ones and zeros')
        end

    end

    function component_cost = cost(obj)
        %Computes the cost of operating the component for the solved
        %thermoelectric power unit based on the given cost parameters
        %and calculated design parameters.

        if obj.onoff == 0
            %'tepowerunit' is turned on

            %Cost of Modules
            if strcmp(obj.options{1},'option1')||...
                    strcmp(obj.options{1},'option3')...
                    ||strcmp(obj.options{1},'option4')...
                    ||strcmp(obj.options{1},'option5')

                %Total Cost of Modules
                cost1(1)=obj.costinfo.specific.module*...
                    obj.module.numperzone*obj.unit.num;

            elseif strcmp(obj.options{1},'option2')
                %Cost of modules if modules are given specific design
                %parameters

                %Module p Leg Cost
                %leg length * leg x_sect_area * num of pairs * num of
                %modules/zone * num of zones
                module_p_leg_volume=obj.module.l_p*obj.module.area_p...
                    *obj.module.num*obj.module.numperzone*obj.unit.num;
                costmodule(1)=...
                    obj.costinfo.specific.module_p_leg_material*...
                    module_p_leg_volume;
```

```
    %Module n Leg Cost
    %leg length * leg x_sect_area * num of pairs * num of
    %modules/zone * num of zones
    module_n_leg_volume=obj.module.l_n*obj.module.area_n...
        *obj.module.num*obj.module.numperzone*obj.unit.num;
    costmodule(2)=...
        obj.costinfo.specific.module_n_leg_material*...
        module_n_leg_volume;

    %Module Ceramic Wafer Cost
    %2 wafers * thickness of wafer * area of module * num
    %of modules/zone * num of zones
    ceramic_volume=2*obj.module.l_cer*...
        obj.module.width^2*obj.module.numperzone*...
        obj.unit.num;
    costmodule(3)=obj.costinfo.specific.module_ceramic*...
        ceramic_volume;

    %Cost Per Leg Pair
    costmodule(4)=...
        obj.costinfo.specific.cost_per_leg_pair*...
        obj.module.num;

    %Module Manufacturing Cost
    costmodule(5)=obj.costinfo.fixed.module_manufac;

    %Total Cost of Modules
    cost1(1)=sum(costmodule);

end


%Cost of Heat Sinks
%Hot Side Fins
if strcmp(obj.options{3},'straightfins_aligned')
    %Material Cost
    fin_volume_h=obj.fins.num_h*obj.fins.l_h*...
        obj.fins.t_h*obj.unit.zone_l+obj.fins.base_t_h*...
        obj.unit.zone_w*obj.unit.zone_l;
    costfin(1)=obj.costinfo.specific.fin_material*...
        fin_volume_h;

    %Fin Manufacturing Cost
    costfin(2)=obj.costinfo.fixed.fin_manufac_h;

elseif strcmp(obj.options{3},'offset_strip_fins')
    %Material Cost
    fin_volume_h=obj.fins.num_h*obj.fins.l_h*...
        obj.fins.t_h*obj.fins.strip_l_h+...
        obj.fins.base_t_h*obj.unit.zone_w*obj.unit.zone_l;
    costfin(1)=obj.costinfo.specific.fin_material*...
        fin_volume_h;

    %Fin Manufacturing Cost
    costfin(2)=obj.costinfo.fixed.fin_manufac_h;
```

```
elseif strcmp(obj.options{3},'pinfins_aligned')||...
        strcmp(obj.options{3},'pinfins_staggered')

    %Material Cost
    fin_volume_h=obj.fins.num_tot_h*obj.fins.l_h*...
        (obj.fins.dia_h^2)*(pi()/4)+obj.fins.base_t_h*...
        obj.unit.zone_w*obj.unit.zone_l;
    costfin(1)=obj.costinfo.specific.fin_material*...
        fin_volume_h;

    %Fin Manufacturing Cost
    costfin(2)=obj.costinfo.fixed.fin_manufac_h;
end

%Cold Side Fins
if strcmp(obj.options{2},'isothermal')||...
        strcmp(obj.options{2},'constantheatrate')
else
    if strcmp(obj.options{4},'straightfins_aligned')
        %Material Cost
        fin_volume_c=obj.fins.num_c*obj.fins.l_c*...
            obj.fins.t_c*obj.unit.zone_l+...
            obj.fins.base_t_c*obj.unit.zone_w*...
            obj.unit.zone_l;
        costfin(3)=obj.costinfo.specific.fin_material*...
            fin_volume_c;

        %Fin Manufacturing Cost
        costfin(4)=obj.costinfo.fixed.fin_manufac_c;

    elseif strcmp(obj.options{4},'offset_strip_fins')
        %Material Cost
        fin_volume_c=obj.fins.num_c*obj.fins.l_c*...
            obj.fins.t_c*obj.fins.strip_l_c+...
            obj.fins.base_t_c*obj.unit.zone_w*...
            obj.unit.zone_l;
        costfin(3)=obj.costinfo.specific.fin_material*...
            fin_volume_c;

        %Fin Manufacturing Cost
        costfin(4)=obj.costinfo.fixed.fin_manufac_c;

    elseif strcmp(obj.options{4},'pinfins_aligned')||...
            strcmp(obj.options{4},'pinfins_staggered')
        %Material Cost
        fin_volume_c=obj.fins.num_tot_c*obj.fins.l_c*...
            (obj.fins.dia_c^2)*(pi()/4)+...
            obj.fins.base_t_c*obj.unit.zone_w*...
            obj.unit.zone_l;
        costfin(3)=obj.costinfo.specific.fin_material*...
            fin_volume_c;

        %Fin Manufacturing Cost
        costfin(4)=obj.costinfo.fixed.fin_manufac_c;
```

```
            end
        end

        %Total Cost of Heat Sinks
        cost1(2)=sum(costfin);


        %Cost of Assembly

        %Cost of Insulation
        %Assumes module and insulation are same thickness
        insulation_volume=obj.unit.zone_insul_area*...
            obj.module.thickness*obj.unit.num;
        costassembly(1)=obj.costinfo.specific.insulation*...
            insulation_volume;

        %Cost Per Zone Area
        costassembly(2)=...
            obj.costinfo.specific.cost_per_zone_area*...
            obj.unit.zone_area*obj.unit.num;

        %Other Costs
        costassembly(3)=obj.costinfo.fixed.other;

        %Fixed Assembly Cost
        costassembly(4)=obj.costinfo.fixed.assembly;

        %Total Cost of Assembly
        cost1(3)=sum(costassembly);

        %All cost related parameters to be used by TEPSS
        component_cost.cost = [sum(cost1);0];
        component_cost.power = [0,-obj.power,0,0,0,obj.power,...
            0,0,0]';
        component_cost.emissions = [0,0,0]';
        component_cost.physcon = 0;

    else
        %All cost related parameters to be used by TEPSS if
        %'tepowerunit' is turned off
        component_cost.cost = [0;0];
        component_cost.power = [0,0,0,0,0,0,0,0,0]';
        component_cost.emissions = [0,0,0]';
        component_cost.physcon = 0;
    end
end

function module_parameters(obj)
    %MODULE_PARAMETERS finds the 3 module performance parameters.
    %The MODULE_PARAMETERS function calculates the 3 module
    %performance parameters based on the selected options and
    %associated input parameters.
    %
    %option1 - The three parameters are provided by the user as
    %predetermined through empirical means.
```

```
%
%option2 - The three parameters are determined through the
%input material properties and geometries. The model assumes 1D
%transport with constant properties.  The leg cross sectional
%area is assumed to be the same for both sides of the leg pair.
%The properties are also assumed to be the same for both legs.
%The electrical resistance is the sum of the resistance of the
%two legs plus the contact resistance at the four interfaces
%times the number of leg pairs. Similarly the thermal
%resistance is simply the sum of the leg pair resistance plus a
%top and bottom layer resistance. Heat spreading is neglected.
%To determine the effective module Seebeck coefficient an
%energy balance was performed.
%E. J. Sandoz-Roszado, "Investigation and development of
%advanced models of thermoelectric generators for power
%generation applications". M.S. Thesis, Department of
%Mechanical Engineering, Rochester Institute of Technology,
%Rochester, NY, USA, 2009.
%
%option3 - The three parameters are found using Thermoelectric
%Cooler manufacturer provided data. This method uses maximum
%temperature difference, hot side temperature, maximum current,
%and maximum voltage.
%Z. Luo, "A Simple Method to Estimate the Physical
%Characteristics of a Thermoelectric Cooler from Vendor
%Datasheets," Electronics Cooling, 2008, Available:
%http://www.electronics-cooling.com/
%
%option4 - The three parameters are found using Thermoelectric
%Cooler manufacturer provided data. This method uses maximum
%temperature difference, hot side temperature, maximum current,
%and maximum heat flow.
%Z. Luo, "A Simple Method to Estimate the Physical
%Characteristics of a Thermoelectric Cooler from Vendor
%Datasheets," Electronics Cooling, 2008, Available:
%http://www.electronics-cooling.com/
%
%option5 - The three parameters are determined from
%Thermoelectric Generator manufacturer provided data.

if strcmp(obj.options{1},'option1')
    %See 'option1' description above
    obj.module.telec_resist;
    obj.module.ttherm_resist;
    obj.module.talpha;
    obj.module.width;
    obj.module.thickness;

elseif strcmp(obj.options{1},'option2')
    %See 'option2' description above
    rho_p=obj.module.rho_p;
    alpha_p=obj.module.alpha_p;
    cond_p=obj.module.k_p;
    L_p=obj.module.l_p;
    A_p=obj.module.area_p;
    rho_n=obj.module.rho_n;
```

```
        alpha_n=obj.module.alpha_n;
        cond_n=obj.module.k_n;
        L_n=obj.module.l_n;
        A_n=obj.module.area_n;
        a_ratio=obj.module.a_ratio;
        num=obj.module.num;
        res_contact=obj.module.contact_resist;
        L_cer=obj.module.l_cer;
        k_cer=obj.module.k_cer;

        %Equations that use p & n type leg thermoelectric
        %parameters
        obj.module.telec_resist=num*rho_p*L_p/A_p+num*rho_n*...
            L_n/A_n+2*num*res_contact/A_p+2*num*res_contact/A_n;
        obj.module.ttherm_resist=1/((num*cond_p*A_p)/L_p+(num*...
            cond_n*A_n)/L_n);
        obj.module.talpha=num*(alpha_p-alpha_n);
        %Assumes module is square
        obj.module.width=sqrt(num*(A_p+A_n)*a_ratio);
        obj.module.thickness=max(L_p,L_n)+2*L_cer;

        %Module Ceramic Thermal Resistance
        obj.module.tcer_resist=L_cer/(num*k_cer*(A_p+A_n)*a_ratio);

    elseif strcmp(obj.options{1}, 'option3')
        %See 'option3' description above
        T_h=obj.module.T_h;
        deltaTmax=obj.module.deltaTmax;
        Imax=obj.module.Imax;
        Vmax=obj.module.Vmax;

        obj.module.talpha=Vmax/T_h;
        obj.module.ttherm_resist=2*T_h*deltaTmax/((T_h-...
            deltaTmax)*Vmax*Imax);
        obj.module.telec_resist=((T_h-deltaTmax)*Vmax)/(T_h*Imax);
        obj.module.width;
        obj.module.thickness;

    elseif strcmp(obj.options{1}, 'option4')
        %See 'option4' description above
        T_h=obj.module.T_h;
        deltaTmax=obj.module.deltaTmax;
        Imax=obj.module.Imax;
        Qmax=obj.module.Qmax;

        obj.module.talpha=2*Qmax/(Imax*(T_h+deltaTmax));
        obj.module.ttherm_resist=((T_h+deltaTmax)*deltaTmax)/...
            ((T_h-deltaTmax)*Qmax);
        obj.module.telec_resist=obj.module.talpha^2*...
            obj.module.ttherm_resist*(T_h-deltaTmax)^2/...
            (2*deltaTmax);
        obj.module.width;
        obj.module.thickness;

    elseif strcmp(obj.options{1}, 'option5')
```

```
        %See 'option5' description above
        V_oc=obj.module.V_oc;
        T_h=obj.module.T_h;
        T_c=obj.module.T_c;
        I_load=obj.module.I_load;
        modq_h=obj.module.q_h;

        obj.module.telec_resist;
        obj.module.talpha=V_oc/(T_h-T_c);
        obj.module.ttherm_resist=(T_h-T_c)/(modq_h-...
            obj.module.talpha*I_load*T_h+0.5*(I_load)^2*...
            obj.module.telec_resist);
        obj.module.width;
        obj.module.thickness;

    end

    %Assume modules are ELECTRICALLY in SERIES
    obj.module.numperzone=obj.unit.series*obj.unit.parallel;
    obj.module.telec_resist_group=obj.module.telec_resist*...
        obj.module.numperzone;
    obj.module.talpha_group=obj.module.talpha*...
        obj.module.numperzone;

    %Thermal resistance for multiple modules in zones
    obj.module.ttherm_resist_group=obj.module.ttherm_resist/...
        obj.module.numperzone;

    if strcmp(obj.options{1}, 'option2')
        obj.module.tcer_resist_group=obj.module.tcer_resist/...
            obj.module.numperzone;
    end
end

function unit_setup(obj)
    %Thermoelectric Power Unit Parameters are determined in this
    %function. Additonal functions are called within this function
    %to complete the thermoelectric power unit setup.

    %Adds module lengths and widths together for all modules in
    %zone. Then finds the area and a length to width ratio. This
    %ratio is assumed to be true for the insulation area by
    %proportionality.
    obj.unit.zone_mod_l=obj.module.width*obj.unit.series;
    obj.unit.zone_mod_w=obj.module.width*obj.unit.parallel;
    obj.unit.zone_mod_area=obj.unit.zone_mod_l*obj.unit.zone_mod_w;
    obj.unit.zone_lw_ratio=obj.unit.zone_mod_l/obj.unit.zone_mod_w;

    %"obj.unit.zone_to_mod_area_ratio" - ratio is the total zone
    %face area to the module area with insulation being the
    %difference in areas.
    obj.unit.zone_area=obj.unit.zone_mod_area*...
        obj.unit.zone_to_mod_area_ratio;
    obj.unit.zone_l=sqrt(obj.unit.zone_lw_ratio*...
        obj.unit.zone_area);
```

```
    obj.unit.zone_w=sqrt((1/obj.unit.zone_lw_ratio)*...
        obj.unit.zone_area);

    obj.unit.zone_insul_area=obj.unit.zone_area-...
        obj.unit.zone_mod_area;

    %Hot Side Fin Calculations
    obj.hot_side_fins_setup;

    %Hot Side Environmental Losses (Used in Fsolve)
    obj.unit.envir_surf_area_h=obj.unit.zone_area+2*...
        obj.unit.zone_l*obj.fins.l_h;
    obj.unit.envir_cond_res_h=obj.unit.shell_t/...
        (obj.unit.shell_k*obj.unit.envir_surf_area_h);
    obj.unit.envir_UA_h=1/(1/(obj.unit.uvalue*...
        obj.unit.envir_surf_area_h)+obj.unit.envir_cond_res_h);

    if strcmp(obj.options{2},'isothermal')||...
            strcmp(obj.options{2},'constantheatrate')
    else
        %Cold Side Fin Calculations
        obj.cold_side_fins_setup;

        %Cold Side Environmental Losses (Used in Fsolve)
        obj.unit.envir_surf_area_c=obj.unit.zone_area+2*...
            obj.unit.zone_l*obj.fins.l_c;
        obj.unit.envir_cond_res_c=obj.unit.shell_t/...
            (obj.unit.shell_k*obj.unit.envir_surf_area_c);
        obj.unit.envir_UA_c=1/(1/(obj.unit.uvalue*...
            obj.unit.envir_surf_area_c)+obj.unit.envir_cond_res_c);
    end

    %Insulation thermal resistance
    obj.unit.insul_therm_res=obj.module.thickness/...
        (obj.unit.insul_k*obj.unit.zone_insul_area);

end

function y = paramcheck(obj)
    %This function checks to ensure that the parameters determined
    %from the input parameters do not create an unrealistic
    %thermoelectric power unit. An unrealistic power unit
    %determined by this function would result in output noting
    %unrealistic conditions and require new input parameters to be
    %defined by the user.

    y=0;

    %Hot Side Fins Check
    if strcmp(obj.options{3},'straightfins_aligned')

        if obj.unit.num_channel_h<1||obj.fins.pitch_h<0||...
                obj.fins.fin_spacing_h<0||obj.fins.l_h<=0
            y=1;
```

```
                    disp('The proposed hot side finned heat exchangers are
unrealistic and will yield inaccurate results.')
                else
                end

            elseif strcmp(obj.options{3},'offset_strip_fins')

                if obj.unit.num_entra_channel_h<1||obj.fins.pitch_h<0||...
                        obj.fins.fin_spacing_h<0||obj.fins.l_h<=0||...
                        obj.fins.rows_num_h<1
                    y=1;
                    disp('The proposed hot side finned heat exchangers are
unrealistic and will yield inaccurate results.')
                else
                end

            elseif strcmp(obj.options{3},'pinfins_aligned')||...
                    strcmp(obj.options{3},'pinfins_staggered')

                if obj.fins.l_h<=0||obj.fins.pinA1_h<=0||...
                        obj.fins.num_rows_h<1||obj.fins.num_columns_h<1
                    y=1;
                    disp('The proposed hot side finned heat exchangers are
unrealistic and will yield inaccurate results.')
                else
                end
            end

            if strcmp(obj.options{2},'isothermal')||...
                    strcmp(obj.options{2},'constantheatrate')

            else

                %Cold Side Fins Check
                if strcmp(obj.options{4},'straightfins_aligned')

                    if obj.unit.num_channel_c<1||obj.fins.pitch_c<0||...
                            obj.fins.fin_spacing_c<0||obj.fins.l_c<=0
                        y=1;
                        disp('The proposed cold side finned heat exchangers
are unrealistic and will yield inaccurate results.')
                    else
                    end

                elseif strcmp(obj.options{4},'offset_strip_fins')

                    if obj.unit.num_entra_channel_c<1||...
                            obj.fins.pitch_c<0||...
                            obj.fins.fin_spacing_c<0||obj.fins.l_c<=0||...
                            obj.fins.rows_num_c<1
                        y=1;
                        disp('The proposed cold side finned heat exchangers
are unrealistic and will yield inaccurate results.')
                    else
                    end
```

```
                elseif strcmp(obj.options{4},'pinfins_aligned')||...
                        strcmp(obj.options{4},'pinfins_staggered')

                    if obj.fins.l_c<=0||obj.fins.pinA1_c<=0||...
                            obj.fins.num_rows_c<1||obj.fins.num_columns_c<1
                        y=1;
                        disp('The proposed cold side finned heat exchangers
are unrealistic and will yield inaccurate results.')
                    else
                    end
                end
            end
        end

        function hot_side_fins_setup(obj)
            %Sets up of the thermoelectric power unit hot side fins.
            %Calculations are based on the type of fins selected by the
            %user.

            %Hot Side Fin Calculations
            if strcmp(obj.options{3},'straightfins_aligned')
                %Rectangular Straight Fins
                %Hot side
                obj.unit.num_channel_h=obj.fins.num_h-1;
                %Pitch of fins assuming outer edges count as fins
                obj.fins.pitch_h=(obj.unit.zone_w-obj.fins.t_h)/...
                    obj.unit.num_channel_h;
                obj.fins.fin_spacing_h=obj.fins.pitch_h-obj.fins.t_h;
                obj.unit.perim_wet_h=2*obj.fins.l_h+2*...
                    obj.fins.fin_spacing_h;
                obj.unit.hyd_dia_h=4*obj.fins.l_h*...
                    obj.fins.fin_spacing_h/obj.unit.perim_wet_h;
                obj.unit.entr_area_h=obj.fins.fin_spacing_h*...
                    obj.fins.l_h*obj.unit.num_channel_h;
                obj.fins.lc_h=obj.fins.l_h+obj.fins.t_h/2;
                obj.fins.face_perim_h=2*(obj.fins.t_h+obj.unit.zone_l);
                obj.fins.x_sect_area_h=obj.fins.t_h*obj.unit.zone_l;
                obj.fins.surf_area_h=2*obj.unit.num_channel_h*...
                    obj.fins.lc_h*obj.unit.zone_l;
                obj.fins.base_surf_area_h=obj.unit.zone_area-...
                    obj.fins.x_sect_area_h*obj.fins.num_h;
                obj.fins.total_surf_area_h=obj.fins.surf_area_h+...
                    obj.fins.base_surf_area_h;


            elseif strcmp(obj.options{3},'offset_strip_fins')
                %Offset Strip Fins
                %Hot Side
                obj.unit.num_entra_channel_h=obj.fins.trans_num_h-1;
                obj.fins.pitch_h=(obj.unit.zone_w-obj.fins.t_h)/...
                    obj.unit.num_entra_channel_h;
                obj.fins.fin_spacing_h=obj.fins.pitch_h-obj.fins.t_h;
                obj.fins.epsil_h=(obj.fins.pitch_h-obj.fins.t_h)/...
                    obj.fins.pitch_h;
                obj.fins.strip_l_h=obj.unit.zone_l/obj.fins.rows_num_h;
```

```
        obj.unit.hyd_dia_h=(4*obj.fins.fin_spacing_h*...
            obj.fins.l_h*obj.fins.strip_l_h)/...
            (2*(obj.fins.fin_spacing_h*obj.fins.strip_l_h+...
            obj.fins.l_h*obj.fins.strip_l_h+obj.fins.t_h*...
            obj.fins.l_h)+obj.fins.t_h*obj.fins.fin_spacing_h);
        obj.unit.entr_area_h=obj.fins.fin_spacing_h*...
            obj.fins.l_h*obj.unit.num_entra_channel_h;
        obj.fins.lc_h=obj.fins.l_h+obj.fins.t_h/2;
        obj.fins.face_perim_h=2*(obj.fins.t_h+obj.fins.strip_l_h);
        obj.fins.x_sect_area_h=obj.fins.t_h*obj.fins.strip_l_h;
        obj.fins.num_h=obj.fins.rows_num_h*obj.fins.trans_num_h-...
            floor(obj.fins.rows_num_h/2);
        obj.fins.surf_area_h=2*obj.fins.num_h*obj.fins.lc_h*...
            obj.fins.strip_l_h;
        obj.fins.base_surf_area_h=obj.unit.zone_area-...
            obj.fins.x_sect_area_h*obj.fins.num_h;
        obj.fins.total_surf_area_h=obj.fins.surf_area_h+...
            obj.fins.base_surf_area_h;


    elseif strcmp(obj.options{3},'pinfins_aligned')||...
            strcmp(obj.options{3},'pinfins_staggered')
        %Pin Fins
        %Hot Side
        obj.fins.lc_h=obj.fins.l_h+obj.fins.dia_h/4;
        obj.fins.pinA1_h=obj.fins.pitch_trans_h-obj.fins.dia_h;
        obj.fins.x_sect_area_h=pi()*(obj.fins.dia_h^2)/4;
        obj.unit.perim_wet_h=pi()*obj.fins.dia_h;
        %floor() rounds down
        obj.fins.num_rows_h=floor((obj.unit.zone_l-...
            obj.fins.dia_h)/obj.fins.pitch_long_h+1);
        obj.fins.num_columns_h=floor((obj.unit.zone_w-...
            obj.fins.dia_h)/obj.fins.pitch_trans_h+1);
        obj.unit.entr_area_h=obj.unit.zone_w*obj.fins.l_h-...
            obj.fins.num_columns_h*obj.fins.l_h*obj.fins.dia_h;
        if strcmp(obj.options{3},'pinfins_aligned')

            obj.fins.num_tot_h=obj.fins.num_columns_h*...
                obj.fins.num_rows_h;

        elseif strcmp(obj.options{3},'pinfins_staggered')

            obj.fins.num_tot_h=obj.fins.num_columns_h*...
                obj.fins.num_rows_h-floor(obj.fins.num_rows_h/2);
            %Additional calculations for staggered pin fins
            obj.fins.pitch_d_h=sqrt(obj.fins.pitch_long_h^2+...
                (obj.fins.pitch_trans_h/2)^2);
            obj.fins.pinA2_h=obj.fins.pitch_d_h-obj.fins.dia_h;
        end
        obj.fins.surf_area_h=pi()*obj.fins.dia_h*obj.fins.lc_h*...
            obj.fins.num_tot_h;
        obj.fins.base_surf_area_h=obj.unit.zone_area-...
            obj.fins.x_sect_area_h*obj.fins.num_tot_h;
        obj.fins.total_surf_area_h=obj.fins.surf_area_h+...
            obj.fins.base_surf_area_h;
```

```
    end

    %Base of hot side fins
    obj.fins.base_therm_res_h=obj.fins.base_t_h/...
        (obj.fins.k_h*obj.unit.zone_area);

end

function cold_side_fins_setup(obj)
    %Sets up of the thermoelectric power unit cold side fins.
    %Calculations are based on the type of fins selected by the
    %user.

    %Cold Side Fin Calculations
    if strcmp(obj.options{4},'straightfins_aligned')
        %Rectangular Straight Fins
        %Cold side
        obj.unit.num_channel_c=obj.fins.num_c-1;
        %Pitch of fins assuming outer edges count as fins
        obj.fins.pitch_c=(obj.unit.zone_w-obj.fins.t_c)/...
            obj.unit.num_channel_c;
        obj.fins.fin_spacing_c=obj.fins.pitch_c-obj.fins.t_c;
        obj.unit.perim_wet_c=2*obj.fins.l_c+2*...
            obj.fins.fin_spacing_c;
        obj.unit.hyd_dia_c=4*obj.fins.l_c*...
            obj.fins.fin_spacing_c/obj.unit.perim_wet_c;
        obj.unit.entr_area_c=obj.fins.fin_spacing_c*...
            obj.fins.l_c*obj.unit.num_channel_c;
        obj.fins.lc_c=obj.fins.l_c+obj.fins.t_c/2;
        obj.fins.face_perim_c=2*(obj.fins.t_c+obj.unit.zone_l);
        obj.fins.x_sect_area_c=obj.fins.t_c*obj.unit.zone_l;
        obj.fins.surf_area_c=2*obj.unit.num_channel_c*...
            obj.fins.lc_c*obj.unit.zone_l;
        obj.fins.base_surf_area_c=obj.unit.zone_area-...
            obj.fins.x_sect_area_c*obj.fins.num_c;
        obj.fins.total_surf_area_c=obj.fins.surf_area_c+...
            obj.fins.base_surf_area_c;


    elseif strcmp(obj.options{4},'offset_strip_fins')
        %Offset Strip Fins
        %Cold Side
        obj.unit.num_entra_channel_c=obj.fins.trans_num_c-1;
        obj.fins.pitch_c=(obj.unit.zone_w-obj.fins.t_c)/...
            obj.unit.num_entra_channel_c;
        obj.fins.fin_spacing_c=obj.fins.pitch_c-obj.fins.t_c;
        obj.fins.epsil_c=(obj.fins.pitch_c-obj.fins.t_c)/...
            obj.fins.pitch_c;
        obj.fins.strip_l_c=obj.unit.zone_l/obj.fins.rows_num_c;
        obj.unit.hyd_dia_c=(4*obj.fins.fin_spacing_c*...
            obj.fins.l_c*obj.fins.strip_l_c)/...
            (2*(obj.fins.fin_spacing_c*obj.fins.strip_l_c+...
            obj.fins.l_c*obj.fins.strip_l_c+obj.fins.t_c*...
            obj.fins.l_c)+obj.fins.t_c*obj.fins.fin_spacing_c);
```

```
    obj.unit.entr_area_c=obj.fins.fin_spacing_c*...
        obj.fins.l_c*obj.unit.num_entra_channel_c;
    obj.fins.lc_c=obj.fins.l_c+obj.fins.t_c/2;
    obj.fins.face_perim_c=2*(obj.fins.t_c+obj.fins.strip_l_c);
    obj.fins.x_sect_area_c=obj.fins.t_c*obj.fins.strip_l_c;
    obj.fins.num_c=obj.fins.rows_num_c*obj.fins.trans_num_c-...
        floor(obj.fins.rows_num_c/2);
    obj.fins.surf_area_c=2*obj.fins.num_c*obj.fins.lc_c*...
        obj.fins.strip_l_c;
    obj.fins.base_surf_area_c=obj.unit.zone_area-...
        obj.fins.x_sect_area_c*obj.fins.num_c;
    obj.fins.total_surf_area_c=obj.fins.surf_area_c+...
        obj.fins.base_surf_area_c;


elseif strcmp(obj.options{4},'pinfins_aligned')||...
        strcmp(obj.options{4},'pinfins_staggered')
    %Pin Fins
    %Cold Side
    obj.fins.lc_c=obj.fins.l_c+obj.fins.dia_c/4;
    obj.fins.pinA1_c=obj.fins.pitch_trans_c-obj.fins.dia_c;
    obj.fins.x_sect_area_c=pi()*(obj.fins.dia_c^2)/4;
    obj.unit.perim_wet_c=pi()*obj.fins.dia_c;
    %floor() rounds down
    obj.fins.num_rows_c=floor((obj.unit.zone_l-...
        obj.fins.dia_c)/obj.fins.pitch_long_c+1);
    obj.fins.num_columns_c=floor((obj.unit.zone_w-...
        obj.fins.dia_c)/obj.fins.pitch_trans_c+1);
    obj.unit.entr_area_c=obj.unit.zone_w*obj.fins.l_c-...
        obj.fins.num_columns_c*obj.fins.l_c*obj.fins.dia_c;
    if strcmp(obj.options{4},'pinfins_aligned')

        obj.fins.num_tot_c=obj.fins.num_columns_c*...
            obj.fins.num_rows_c;

    elseif strcmp(obj.options{4},'pinfins_staggered')

        obj.fins.num_tot_c=obj.fins.num_columns_c*...
            obj.fins.num_rows_c-floor(obj.fins.num_rows_c/2);
        %Additional calculations for staggered pin fins
        obj.fins.pitch_d_c=sqrt(obj.fins.pitch_long_c^2+...
            (obj.fins.pitch_trans_c/2)^2);
        obj.fins.pinA2_c=obj.fins.pitch_d_c-obj.fins.dia_c;
    end
    obj.fins.surf_area_c=pi()*obj.fins.dia_c*obj.fins.lc_c*...
        obj.fins.num_tot_c;
    obj.fins.base_surf_area_c=obj.unit.zone_area-...
        obj.fins.x_sect_area_c*obj.fins.num_tot_c;
    obj.fins.total_surf_area_c=obj.fins.surf_area_c+...
        obj.fins.base_surf_area_c;


end

%Base of cold side fins
```

```
    obj.fins.base_therm_res_c=obj.fins.base_t_c/(obj.fins.k_c*...
        obj.unit.zone_area);

end

function [fin_res press_loss cp] = precalc(obj,fluidin,fluidout,...
        hotcold,option_fin)
    %This function calculates the pressure drop and thermal
    %resistance for a given side of the power unit.  Assumes fully
    %developed flow.

    mdot=fluidin.mdot;

    %Property Values
    %FluidProp is used for finding thermophysical parameters
    %FluidProp is used for fluidin & fluidout and averaged
    cpin=fluidin.getprop('HeatCapP','Ph',...
        fluidin.press,fluidin.enthalpy);
    densityin=fluidin.getprop('Density','Ph',...
        fluidin.press,fluidin.enthalpy);
    viscin=fluidin.getprop('Viscosity','Ph',...
        fluidin.press,fluidin.enthalpy);
    kin=fluidin.getprop('ThermCond','Ph',...
        fluidin.press,fluidin.enthalpy);

    cpout=fluidout.getprop('HeatCapP','Ph',...
        fluidout.press,fluidout.enthalpy);
    densityout=fluidout.getprop('Density','Ph',...
        fluidout.press,fluidout.enthalpy);
    viscout=fluidout.getprop('Viscosity','Ph',...
        fluidout.press,fluidout.enthalpy);
    kout=fluidout.getprop('ThermCond','Ph',...
        fluidout.press,fluidout.enthalpy);

    cp=(cpin+cpout)/2;
    density=(densityin+densityout)/2;
    visc=(viscin+viscout)/2;
    k=(kin+kout)/2;

    %Calls other functions based on fin type. Returns value. for fin
    %thermal resistance and pressure drop across fins.
    if strcmp(option_fin,'straightfins_aligned')

        [fin_res, press_loss] = obj.straightfins(mdot,cp,...
            density,visc,k,fluidin,hotcold);

    elseif strcmp(option_fin,'offset_strip_fins')

        [fin_res, press_loss] = obj.offsetstripfins(mdot,cp,...
            density,visc,k,hotcold);

    elseif strcmp(option_fin,'pinfins_aligned')||...
            strcmp(option_fin,'pinfins_staggered')
```

```
        [fin_res, press_loss] = obj.pinfins(mdot,cp,...
            density,visc,k,fluidin,hotcold,option_fin);

    end
end

function [fin_res press_loss] = straightfins(obj,mdot,cp,...
        density,visc,k,fluidin,hotcold)
    %Function for determining rectangular straight fins performance
    %parameters. Final results are thermal resistance of the fins
    %and pressure drop across the fins.

    if strcmp(hotcold,'hot')
        perim_wet=obj.unit.perim_wet_h;
        num_channel=obj.unit.num_channel_h;
        hyd_dia=obj.unit.hyd_dia_h;
        fins_k=obj.fins.k_h;
        fins_lc=obj.fins.lc_h;
        fins_surf_area=obj.fins.surf_area_h;
        base_therm_res=obj.fins.base_therm_res_h;
        entr_area=obj.unit.entr_area_h;
        perim=obj.fins.face_perim_h;
        x_sect_area=obj.fins.x_sect_area_h;
        total_surf_area=obj.fins.total_surf_area_h;
        temp_s=obj.surf_temp_hot;
        fins_l=obj.fins.l_h;
        fins_space=obj.fins.fin_spacing_h;
        %For Heat Spreading
        fins_base_t=obj.fins.base_t_h;

    elseif strcmp(hotcold,'cold')
        perim_wet=obj.unit.perim_wet_c;
        num_channel=obj.unit.num_channel_c;
        hyd_dia=obj.unit.hyd_dia_c;
        fins_k=obj.fins.k_c;
        fins_lc=obj.fins.lc_c;
        fins_surf_area=obj.fins.surf_area_c;
        base_therm_res=obj.fins.base_therm_res_c;
        entr_area=obj.unit.entr_area_c;
        perim=obj.fins.face_perim_c;
        x_sect_area=obj.fins.x_sect_area_c;
        total_surf_area=obj.fins.total_surf_area_c;
        temp_s=obj.surf_temp_cold;
        fins_l=obj.fins.l_c;
        fins_space=obj.fins.fin_spacing_c;
        %For Heat Spreading
        fins_base_t=obj.fins.base_t_c;
    end

    %Reynolds number
    Re=4*mdot/(visc*perim_wet*num_channel);

    %Prandtl number
    Pr=cp*visc/k;
```

```
%Friction factor
if Re<3000

    if fins_l>=fins_space
        aspect_ratio=fins_l/fins_space;

    elseif fins_space>fins_l
        aspect_ratio=fins_space/fins_l;
    end

    if aspect_ratio<=8
        f_x_Re=-0.4673*aspect_ratio^2+7.8663*aspect_ratio...
            +49.006;

    elseif aspect_ratio>8
        f_x_Re=96;
    end

    fric=f_x_Re/Re;

    %Surface viscosity
    visc_s=fluidin.getprop('Viscosity','PT',...
        fluidin.press,temp_s);

    %Nusselt number
    Nu=1.86*(Re*Pr/(obj.unit.zone_l/hyd_dia))^(1/3)*...
        (visc/visc_s)^0.14;
    %E. N. Sieder and G. E. Tate, "Heat Transfer and Pressure
    %Drop of Liquids in Tubes," Ind. Eng. Chem., vol. 28,
    %1429-1436, 1936.
    %Laminar, combined entry, 0.6<=Pr<=5,
    %0.0044<=(visc/visc_s)<=9.75, uniform temp_s
    %Heat Transfer Book Page 513, eqn (8.57)

    if Nu<3.66
        Nu=3.66;
    end

elseif 3000<=Re && Re<=5*10^6

    fric=(0.790*log(Re)-1.64)^(-2);
    %Turbulent, fully developed, 3000<=Re<=5*10^6
    %B. S. Petukhov, "Heat transfer and friction in turbulent
    %pipe flow with variable physical properties," in T. F.
    %Irvine and J. P. Hartnett, (Editors)., Advances in Heat
    %Transfer, vol. 6, Academic Press, New York, 1970.

    %Nusselt number
    Nu=(fric/8)*(Re-1000)*Pr/(1+12.7*((fric/8)^(1/2))*...
        (Pr^(2/3)-1));
    %V. Gnielinski, "New Equations for Heat and Mass Transfer
    %in Turbulent Pipe and Channel Flow," International
    %Chemical Engineering, vol. 16, pp. 359-368, 1976.
    %Turbulent, fully developed, 0.5<=Pr<=2000,
    %3000<=Re<=5*10^6, (L/D)>=10
```

```
            else
                disp('The Reynolds Number is outside of the acceptable
range.')
                disp('Implement another correlation for Nusselt Number that
is appropriate for Re>5*10^6.')
            end

            %Convective coefficient (h)
            conv_coeff=(Nu*k/hyd_dia);

            %Calculate m for uniform cross sectional rect. fin
            m=sqrt(conv_coeff*perim/(fins_k*x_sect_area));

            %Efficiency of a single fin
            fin_eff=tanh(m*fins_lc)/(m*fins_lc);

            %Overall surface efficiency
            overall_eff=1-(fins_surf_area/total_surf_area)*(1-fin_eff);

            %Effective area
            A_eff=overall_eff*total_surf_area;

            %Effective convective coefficient
            h_eff=conv_coeff*(A_eff/(obj.unit.zone_w*obj.unit.zone_l));

            %Spreading resistance from outside function
            [R_sp] = obj.spreading_res(h_eff,fins_base_t,fins_k);

            %Thermal resistance of the fin arrays [K/W]
            fin_res=((1/(overall_eff*conv_coeff*total_surf_area))+R_sp+...
                base_therm_res);

            %Velocity of air through channels
            vel=mdot/(entr_area*density);

            %Major head losses through all channels [N/m^2=Pa]
            press_loss=fric*density*vel^2/(2*hyd_dia)*obj.unit.zone_l;

        end

        function [fin_res press_loss] = offsetstripfins(obj,mdot,cp,...
                density,visc,k,hotcold)
            %Function for determining offset strip fins performance
            %parameters. Final results are thermal resistance of the fins
            %and pressure drop across the fins.

            if strcmp(hotcold,'hot')
                fins_t=obj.fins.t_h;
                fins_l=obj.fins.l_h;
                fins_lc=obj.fins.lc_h;
                fins_strip_l=obj.fins.strip_l_h;
                fin_spacing=obj.fins.fin_spacing_h;
                entr_area=obj.unit.entr_area_h;
```

```
    hyd_dia=obj.unit.hyd_dia_h;
    epsil=obj.fins.epsil_h;
    fins_k=obj.fins.k_h;
    perim=obj.fins.face_perim_h;
    x_sect_area=obj.fins.x_sect_area_h;
    fins_surf_area=obj.fins.surf_area_h;
    total_surf_area=obj.fins.total_surf_area_h;
    base_therm_res=obj.fins.base_therm_res_h;
    %For Heat Spreading
    fins_base_t=obj.fins.base_t_h;

elseif strcmp(hotcold,'cold')
    fins_t=obj.fins.t_c;
    fins_l=obj.fins.l_c;
    fins_lc=obj.fins.lc_c;
    fins_strip_l=obj.fins.strip_l_c;
    fin_spacing=obj.fins.fin_spacing_c;
    entr_area=obj.unit.entr_area_c;
    hyd_dia=obj.unit.hyd_dia_c;
    epsil=obj.fins.epsil_c;
    fins_k=obj.fins.k_c;
    perim=obj.fins.face_perim_c;
    x_sect_area=obj.fins.x_sect_area_c;
    fins_surf_area=obj.fins.surf_area_c;
    total_surf_area=obj.fins.total_surf_area_c;
    base_therm_res=obj.fins.base_therm_res_c;
    %For Heat Spreading
    fins_base_t=obj.fins.base_t_c;
end

%Prandtl number
Pr=cp*visc/k;

%Velocity
vel=mdot/(entr_area*density);

%Reynolds number
Re=density*vel*hyd_dia/visc;

%j-factor correlation
%R. M. Manglik and A. E. Bergles, "Heat Transfer and Pressure
%Drop Correlations for the Rectangular Offset Strip Fin Compact
%Heat Exchanger," Experimental Thermal and Fluid Sciences,
%vol. 10, no. 2, pp. 171-180, 1995.
j=(0.6522*(Re^-0.5403))*((fin_spacing/fins_l)^-0.1541)*...
    ((fins_t/fins_strip_l)^0.1499)*...
    ((fins_t/fin_spacing)^-0.0678)*...
    (1+5.269*10^-5*Re^1.340*((fin_spacing/fins_l)^0.504)*...
    ((fins_t/fins_strip_l)^0.456)*...
    ((fins_t/fin_spacing)^-1.055))^0.1;

%Convective coefficient based on LMTD
h_lmtd=j*(density*vel*cp)*Pr^(-2/3);

%NTU
```

```
NTU=(h_lmtd*total_surf_area)/(mdot*cp);

%Convective coefficient (h)
conv_coeff=(mdot*cp/total_surf_area)*(1-exp(-NTU));

%Calculate m for uniform cross sectional rect. fin
m=sqrt(conv_coeff*perim/(fins_k*x_sect_area));

%Efficiency of a single fin
fin_eff=tanh(m*fins_lc)/(m*fins_lc);

%Overall surface efficiency
overall_eff=1-(fins_surf_area/total_surf_area)*(1-fin_eff);

%Effective area
A_eff=overall_eff*total_surf_area;

%Effective convective coefficient
h_eff=conv_coeff*(A_eff/(obj.unit.zone_w*obj.unit.zone_l));

%Spreading resistance from outside function
[R_sp] = obj.spreading_res(h_eff,fins_base_t,fins_k);

%Thermal resistance of the fin arrays [K/W]
fin_res=(1/(overall_eff*conv_coeff*total_surf_area))+R_sp+...
    base_therm_res;

%W. M. Kays and A. L. London, Compact Heat Exchangers,
%3rd ed., McGraw-Hill, New York, 1984.
%Originally found because of:
%S.Y. Kim and R.L. Webb, Analysis of Convective Thermal
%Resistance in Ducted Fan-Heat Sinks, IEEE Transactions on
%Components and Packaging Technologies, Vol. 29, No. 3,
%September 2006

%Inlet loss coefficient (Kc)
inlet_loss_coeff=(0.4+0.03775*epsil-0.4362*epsil^2)/...
    obj.unit.num;

%Exit loss coefficient (Ke)
exit_loss_coeff=((1-epsil)^2)/obj.unit.num;

%Friction factor
%R. M. Manglik and A. E. Bergles, "Heat Transfer and Pressure
%Drop Correlations for the Rectangular Offset Strip Fin Compact
%Heat Exchanger," Experimental Thermal and Fluid Sciences,
%vol. 10, no. 2, pp. 171-180, 1995.
fric=(9.6243*(Re^-0.7422))*((fin_spacing/fins_l)^-0.1856)*...
    ((fins_t/fins_strip_l)^0.3053)*...
    ((fins_t/fin_spacing)^-0.2659)*...
    (1+7.669*10^-8*Re^4.429*((fin_spacing/fins_l)^0.920)*...
    ((fins_t/fins_strip_l)^3.767)*...
    ((fins_t/fin_spacing)^0.236))^0.1;
```

```
    %Pressure loss
    press_loss=(inlet_loss_coeff+exit_loss_coeff+4*fric*...
        obj.unit.zone_1/hyd_dia)*0.5*density*vel^2;

end

function [fin_res press_loss] = pinfins(obj,mdot,cp,...
        density,visc,k,fluidin,hotcold,option_fin)
    %Function for determining aligned and staggered fins
    %performance parameters. Final results are thermal resistance
    %of the fins and pressure drop across the fins.

    if strcmp(hotcold,'hot')
        fins_dia=obj.fins.dia_h;
        pitch_trans=obj.fins.pitch_trans_h;
        pitch_long=obj.fins.pitch_long_h;
        num_rows=obj.fins.num_rows_h;
        entr_area=obj.unit.entr_area_h;
        fins_lc=obj.fins.lc_h;
        fins_k=obj.fins.k_h;
        fins_surf_area=obj.fins.surf_area_h;
        total_surf_area=obj.fins.total_surf_area_h;
        perim=obj.unit.perim_wet_h;
        x_sect_area=obj.fins.x_sect_area_h;
        base_therm_res=obj.fins.base_therm_res_h;
        pinA1=obj.fins.pinA1_h;
        if strcmp(option_fin,'pinfins_aligned')
        elseif strcmp(option_fin,'pinfins_staggered')
            pitch_d=obj.fins.pitch_d_h;
            pinA2=obj.fins.pinA2_h;
        end
        temp_s=obj.surf_temp_hot;
        %For Heat Spreading
        fins_base_t=obj.fins.base_t_h;

    elseif strcmp(hotcold,'cold')
        fins_dia=obj.fins.dia_c;
        pitch_trans=obj.fins.pitch_trans_c;
        pitch_long=obj.fins.pitch_long_c;
        num_rows=obj.fins.num_rows_c;
        entr_area=obj.unit.entr_area_c;
        fins_lc=obj.fins.lc_c;
        fins_k=obj.fins.k_c;
        fins_surf_area=obj.fins.surf_area_c;
        total_surf_area=obj.fins.total_surf_area_c;
        perim=obj.unit.perim_wet_c;
        x_sect_area=obj.fins.x_sect_area_c;
        base_therm_res=obj.fins.base_therm_res_c;
        pinA1=obj.fins.pinA1_c;
        if strcmp(option_fin,'pinfins_aligned')
        elseif strcmp(option_fin,'pinfins_staggered')
            pitch_d=obj.fins.pitch_d_c;
            pinA2=obj.fins.pinA2_c;
        end
        temp_s=obj.surf_temp_cold;
        %For Heat Spreading
```

```
            fins_base_t=obj.fins.base_t_c;
        end

        %Prandtl number
        Pr=cp*visc/k;

        enthalpy=fluidin.getprop('Enthalpy','PT',fluidin.press,temp_s);

        %Surface Temp Properties for Surface Prandtl Number
        %Assumes Ts=Th or Ts=Tc for hot or cold side, respectively
        cp_s=fluidin.getprop('HeatCapP','Ph',fluidin.press,enthalpy);
        visc_s=fluidin.getprop('Viscosity','Ph',...
            fluidin.press,enthalpy);
        k_s=fluidin.getprop('ThermCond','Ph',fluidin.press,enthalpy);

        %Surface Prandtl number
        Pr_s=cp_s*visc_s/k_s;

        %Velocity
        vel=mdot/(density*entr_area);

        if strcmp(option_fin,'pinfins_aligned')
            %Maximum Velocity for Aligned Array
            vel_max=(pitch_trans/pinA1)*vel;

        elseif strcmp(option_fin,'pinfins_staggered')
            %Maximum Velocity for Staggered Array
            check=(pitch_trans+fins_dia)/2;

            if pitch_d<check
                %Vmax occurs at A2
                vel_max=(pitch_trans/(2*pinA2))*vel;

            elseif pitch_d>=check
                %Vmax occurs at A1
                vel_max=(pitch_trans/pinA1)*vel;
            end
        end

        %Maximum Reynolds Number
        Re_max=density*vel_max*fins_dia/visc;

        %Find constants C & m for Nusselt number
        if strcmp(option_fin,'pinfins_aligned')
            if 10<=Re_max && Re_max<10^2
                C=0.80;
                m=0.40;
            elseif 10^2<=Re_max && Re_max<10^3
                %Approximate as a single (isolated) cylinder
                %Average Nusselt Number is found below
            elseif 10^3<=Re_max && Re_max<2*10^5
                abc=pitch_trans/pitch_long;
                if abc<0.7
                    disp('Heat transfer is inefficient and aligned tubes
should not be used.')
```

```
                    return
                elseif abc>=0.7
                    C=0.27;
                    m=0.63;
                end
            elseif 2*10^5<=Re_max && Re_max<=2*10^6
                C=0.021;
                m=0.84;
            else
                disp('Reynolds Number is too high for this Nusselt Number
model.')
                return
            end

        elseif strcmp(option_fin,'pinfins_staggered')
            if 10<=Re_max && Re_max<10^2
                C=0.90;
                m=0.40;
            elseif 10^2<=Re_max && Re_max<10^3
                %Approximate as a single (isolated) cylinder
                %Average Nusselt Number is found below
            elseif 10^3<=Re_max && Re_max<2*10^5
                abc=pitch_trans/pitch_long;
                if abc<2
                    C=0.35*abc^0.2;
                    m=0.60;
                elseif abc>=2
                    C=0.40;
                    m=0.60;
                end
            elseif 2*10^5<=Re_max && Re_max<=2*10^6
                C=0.022;
                m=0.84;
            end
        end

        %Nusselt number
        if Re_max<10^2 || Re_max>=10^3
            %Find constants C2 for Nusselt number
            num_rows_tot=num_rows*obj.unit.num;
            if num_rows_tot>=20
                C2=1;
            elseif num_rows_tot<20
                %Table 7.8 page 440 Heat Transfer book
                %F. P. Incropera, D. P. DeWitt, T. L. Bergman, and A.
                %S. Lavine, Fundamentals of Heat and Mass Transfer,
                %6th ed.  Hoboken, NJ: John Wiley & Sons, Inc., 2007.
                %Interpolated values for 6,8,9,11,12,14,15,17,18,19
                cor_fac=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,...
                    18,19;0.70,0.80,0.86,0.90,0.92,0.935,0.95,...
                    0.9567,0.9633,0.97,0.9733,0.9767,0.98,0.9833,...
                    0.9867,0.99,0.9925,0.995,0.9975;0.64,0.76,0.84,...
                    0.89,0.92,0.935,0.95,0.9567,0.9633,0.97,0.9733,...
                    0.9767,0.98,0.9833,0.9867,0.99,0.9925,0.995,...
                    0.9975];
```

```
            for i=1:size(cor_fac,2)
                if strcmp(option_fin,'pinfins_aligned')
                    if num_rows_tot==cor_fac(1,i)
                        C2=cor_fac(2,i);
                    end
                elseif strcmp(option_fin,'pinfins_staggered')
                    if num_rows_tot==cor_fac(1,i)
                        C2=cor_fac(3,i);
                    end
                end
            end
        end

        %Average Nusselt Number
        %Assumes external flow
        %F. P. Incropera, D. P. DeWitt, T. L. Bergman, and A. S.
        %Lavine, Fundamentals of Heat and Mass Transfer, 6th ed.
        %Hoboken, NJ: John Wiley & Sons, Inc., 2007.
        %Conditions: Average, T_avg, 1000<=Re<=2*10^6,
        %0.7<=Pr<=500
        Nu_avg=C*C2*Re_max^m*Pr^0.36*(Pr/Pr_s)^(0.25);

    elseif 10^2<=Re_max && Re_max<10^3
        %Approximate as a single (isolated) cylinder
        %Average Nusselt Number
        %Assumes external flow
        %F. P. Incropera, D. P. DeWitt, T. L. Bergman, and A. S.
        %Lavine, Fundamentals of Heat and Mass Transfer, 6th ed.
        %Hoboken, NJ: John Wiley & Sons, Inc., 2007.
        %Conditions: Average, T_film, Re*Pr>=0.2
        Nu_avg=0.3+((0.62*(Re_max^0.5)*(Pr^(1/3)))/((1+...
            ((0.4/Pr)^(2/3)))^0.25))*((1+...
            ((Re_max/282000)^(0.625)))^(0.8));
    end

    %Convective Coefficient (h)
    conv_coeff=Nu_avg*k/fins_dia;

    %Calculate m for uniform cross sectional rect. fin
    m1=sqrt(conv_coeff*perim/(fins_k*x_sect_area));

    %Efficiency of a single fin
    fin_eff=tanh(m1*fins_lc)/(m1*fins_lc);

    %Overall surface efficiency
    overall_eff=1-(fins_surf_area/total_surf_area)*(1-fin_eff);

    %Effective area
    A_eff=overall_eff*total_surf_area;

    %Effective convective coefficient
    h_eff=conv_coeff*(A_eff/(obj.unit.zone_w*obj.unit.zone_l));

    %Spreading resistance from outside function
    [R_sp] = obj.spreading_res(h_eff,fins_base_t,fins_k);
```

```
            %Thermal resistance of the fin arrays [K/W]
            fin_res=(1/(overall_eff*conv_coeff*total_surf_area))+...
                R_sp+base_therm_res;

            %Hagen Number
            %Martin,H., 2002, The generalized Leveque equation and its
            %practical use for the prediction of heat and mass transfer
            %rates from pressure drop, Chem. Eng. Sci., Vol. 57, pp.
            %3217-3223.
            %Originally found in Fundamentals of Heat Exchanger Design
            %by Ramesh K. Shah and Dusan P. Sekulic on pages 512-513
            %Aligned and Staggered: 1<Re_max<300000 and num_rows>=5
            %Aligned: 1.25<=nd_pitch_t<=3.0 and 1.2<=nd_pitch_l<=3.0
            %Staggered: 1.25<=nd_pitch_t<=3.0, 0.6<=nd_pitch_l<=3.0,
            %and nd_pitch_d>=1.25
            %Experimental data for this correlation had
            %7.9<=fins_dia<=73mm

            %Non-dimensionalize pitches to diameter of pin fin
            nd_pitch_t=pitch_trans/fins_dia;
            nd_pitch_l=pitch_long/fins_dia;
            if strcmp(option_fin,'pinfins_aligned')
            elseif strcmp(option_fin,'pinfins_staggered')
                nd_pitch_d=pitch_d/fins_dia;
            end

            if strcmp(option_fin,'pinfins_aligned')
                Hg_lam=140*Re_max*((nd_pitch_l^0.5-0.6)^2+0.75)/...
                    ((nd_pitch_t^1.6)*(4*nd_pitch_t*nd_pitch_l/pi()-1));

            elseif strcmp(option_fin,'pinfins_staggered')
                if pitch_d<=check
                    %Vmax occurs at A2
                    Hg_lam=140*Re_max*((nd_pitch_l^0.5-0.6)^2+0.75)/(...
                        (nd_pitch_d^1.6)*(4*nd_pitch_t*nd_pitch_l/pi()-1));

                elseif pitch_d>check
                    %Vmax occurs at A1
                    Hg_lam=140*Re_max*((nd_pitch_l^0.5-0.6)^2+0.75)/(...
                        (nd_pitch_t^1.6)*(4*nd_pitch_t*nd_pitch_l/pi()-1));
                end
            end

            if num_rows_tot<5
                %There isn't a correlation for phi_tn for less than 5
                %rows of pin fins
                disp('There are less than 5 rows of pin fins in the entire
heat exchanger for which this analysis cannot be completed.')
                return

            elseif 5<=num_rows_tot && num_rows_tot<=10
                if strcmp(option_fin,'pinfins_aligned')
                    phi_tn=((1/(2*nd_pitch_t^2))*(1/num_rows_tot-1/10))/...
                        obj.unit.num;
```

```
    elseif strcmp(option_fin,'pinfins_staggered')
        if pitch_d<=check
            %Vmax occurs at A2
            phi_tn=(2*((nd_pitch_d-1)/(nd_pitch_t*...
                (nd_pitch_t-1)))*(1/num_rows_tot-1/10))/...
                obj.unit.num;

        elseif pitch_d>check
            %Vmax occurs at A1
            phi_tn=((1/(2*nd_pitch_t^2))*(1/num_rows_tot-...
                1/10))/obj.unit.num;
        end
    end
elseif num_rows_tot>10
    phi_tn=0;
end

if strcmp(option_fin,'pinfins_aligned')
    Hg_turb=((0.11+(0.6*(1-0.94/nd_pitch_l)^0.6)/...
        ((nd_pitch_t-0.85)^1.3))*10^(0.47*(nd_pitch_l/...
        nd_pitch_t-1.5))+0.015*(nd_pitch_t-1)*...
        (nd_pitch_l-1))*Re_max^(2-0.1*(nd_pitch_l/...
        nd_pitch_t))+phi_tn*Re_max^2;

elseif strcmp(option_fin,'pinfins_staggered')
    if Re_max<=250000
        Hg_turb=((1.25+0.6/((nd_pitch_t-0.85)^1.08))+(0.2*...
            (nd_pitch_l/nd_pitch_t-1)^3)-(0.005*(nd_pitch_t/...
            nd_pitch_l-1)^3))*Re_max^1.75+phi_tn*Re_max^2;

    elseif Re_max>250000
        Hg_turb=(((1.25+0.6/((nd_pitch_t-0.85)^1.08))+(0.2*...
            (nd_pitch_l/nd_pitch_t-1)^3)-(0.005*(nd_pitch_t/...
            nd_pitch_l-1)^3))*Re_max^1.75+phi_tn*Re_max^2)*...
            (1+(Re_max-250000)/325000);
    end
end

if strcmp(option_fin,'pinfins_aligned')
    Hagen=Hg_lam+Hg_turb*(1-exp(1-(Re_max+1000)/2000));
elseif strcmp(option_fin,'pinfins_staggered')
    Hagen=Hg_lam+Hg_turb*(1-exp(1-(Re_max+200)/1000));
end

%Pressure losses
if strcmp(option_fin,'pinfins_aligned')
    press_loss=(visc^2/density)*(num_rows/fins_dia^2)*Hagen;

elseif strcmp(option_fin,'pinfins_staggered')
    if pitch_d<=check
        %Vmax occurs at A2
        press_loss=(visc^2/density)*...
            ((num_rows-1)/fins_dia^2)*Hagen;
    elseif pitch_d>check
```

```
                    %Vmax occurs at A1
                    press_loss=(visc^2/density)*...
                        (num_rows/fins_dia^2)*Hagen;
                end
            end
        end

        function [R_sp_tot] = spreading_res(obj,h,t,k)
            %Three dimensional thermal spreading resistance is found in
            %this function based on the inputs provided from the respective
            %fin type function. The spreading resistance value is
            %determined and provided back to the respective fin type
            %calculations to determine the overall fin resistance.

            %Breaks up a zone into sections based on how many modules are
            %across the zone
            a=obj.unit.zone_w/obj.unit.parallel;

            %Breaks up a zone into sections based on how many modules are
            %along the zone
            b=obj.unit.zone_l/obj.unit.series;

            %Size of one module
            deltax=obj.module.width;
            deltay=obj.module.width; %Module is assumed to be square - this
may change later

            %Non-dimensional variables
            rho=a/b;
            alpha=deltax/a;
            beta=deltay/a;
            tau=t/a;

            NN=10/alpha;

            %Biot number
            biot=h*a/k;
            bt=biot*tau;

            %Dimensionless Spreading Resistance
            %G.E. Ellison, Maximum Thermal Spreading Resistance for
            %Rectangular Sources and Plates With Nonunity Aspect Ratios,
            %IEEE Transactions on Components and Packaging Technologies,
            %Vol. 26, No. 2, June 2003.
            Am=0;
            for ll=1:NN

                Am=Am+(1/ll^3)*((sin(ll*pi*alpha))^2)*((1+(bt/(2*ll*pi*...
                    tau))*tanh(2*ll*pi*tau))/(bt/(2*ll*pi*tau)+...
                    tanh(2*ll*pi*tau)));

            end

            A=(rho/((pi^3)*alpha))*sqrt(beta/alpha)*Am;
```

```
    Bm=0;
    for mm=1:NN

        Bm=Bm+(1/mm^3)*((sin(mm*pi*beta*rho))^2)*((1+(bt/(2*mm*...
            pi*tau*rho))*tanh(2*mm*pi*tau*rho))/((bt/(2*mm*pi*...
            tau*rho))+tanh(2*mm*pi*tau*rho)));

    end

    B=(1/((rho^2)*(pi^3)*beta))*sqrt(alpha/beta)*Bm;


    Clm=0;
    for ll=1:NN

        for mm=1:NN

            Clm=Clm+(1/((ll^2)*(mm^2)))*((sin(ll*pi*alpha))^2)*...
                ((sin(mm*pi*beta*rho))^2)*((1+(bt/(2*pi*tau*...
                sqrt(ll^2+mm^2*rho^2)))*tanh(2*pi*...
                sqrt(ll^2+mm^2*rho^2)*tau))/((2*pi*...
                sqrt(ll^2+mm^2*rho^2))*(((bt*2*pi*tau)/...
                (sqrt(ll^2+mm^2*rho^2)))+tanh(2*pi*tau*...
                sqrt(ll^2+mm^2*rho^2)))));

        end

    end

    C=(4/((pi^4)*rho*alpha*beta*sqrt(alpha*beta)))*Clm;

    %Non-dimensional spreading resistance
    psi_sp=A+B+C;

    %Spreading resistance for a single module in a zone
    R_sp=psi_sp/(k*sqrt(deltax*deltay));

    %Spreading resistance for all modules in a zone
    %All modules are thermally in parallel
    R_sp_tot=R_sp/obj.module.numperzone;

end

function F = solve_zone(obj,x,tempin_h,tempout_c,mdotin_h,mdotin_c)
    %This function solves a zone for the counter flow
    %thermoelectric power unit and then solves successive zones
    %until the entire power unit is solved for. Inputs are received
    %from the other functions and this function is called for by
    %the compute function.

    %Define local variables
```

```
alpha = obj.module.talpha_group;
therm_res = obj.module.ttherm_resist_group;
elec_res = obj.module.telec_resist_group;
load_res=obj.module.zone_load_res;
insul_therm_res=obj.unit.insul_therm_res;
UA_h=obj.unit.UA_h;
UA_c=obj.unit.UA_c;
envir_UA_h=obj.unit.envir_UA_h;
envir_UA_c=obj.unit.envir_UA_c;
envir_temp=obj.unit.envir_temp;

if strcmp(obj.options{1},'option1')||...
        strcmp(obj.options{1},'option3')||...
        strcmp(obj.options{1},'option4')||...
        strcmp(obj.options{1},'option5')

    %Define unknown array
    power1=x(1);
    q_h1=x(2);
    q_c1=x(3);
    te_temp_h=x(4);
    te_temp_c=x(5);
    tempout_h=x(6);
    tempin_c=x(7);
    deltaT_lm_h=x(8);
    deltaT_lm_c=x(9);
    envir_qloss_h=x(10);
    envir_qloss_c=x(11);
    q_ins1=x(12);

    %System of nonlinear equations
    F= [power1-q_h1+q_c1; ...

    alpha*(alpha*(te_temp_h-te_temp_c)/(load_res+...
    elec_res))*te_temp_h+1/therm_res*(te_temp_h-...
    te_temp_c)-0.5*elec_res*(alpha*(te_temp_h-...
    te_temp_c)/(load_res+elec_res))^2-q_h1;...

    alpha*(alpha*(te_temp_h-te_temp_c)/(load_res+...
    elec_res))*te_temp_c+1/therm_res*(te_temp_h-...
    te_temp_c)+0.5*elec_res*(alpha*(te_temp_h-...
    te_temp_c)/(load_res+elec_res))^2-q_c1;...

    mdotin_h*obj.specheat_h*(tempin_h-tempout_h)-...
    UA_h*deltaT_lm_h-envir_qloss_h;...

    mdotin_c*obj.specheat_c*(tempout_c-tempin_c)-...
    UA_c*deltaT_lm_c+envir_qloss_c;...

    q_ins1-(te_temp_h-te_temp_c)/insul_therm_res;...

    UA_h*deltaT_lm_h-q_h1-q_ins1;...

    UA_c*deltaT_lm_c-q_c1-q_ins1;...
```

```
        (tempin_h-tempout_h)/log((abs(tempin_h-te_temp_h)/...
        abs(tempout_h-te_temp_h)))-deltaT_lm_h;...

        (tempin_c-tempout_c)/log((abs(te_temp_c-tempout_c)/...
        abs(te_temp_c-tempin_c)))-deltaT_lm_c;...

        envir_UA_h*(tempin_h-tempout_h)/log((abs(tempin_h-...
        envir_temp)/abs(tempout_h-envir_temp)))-envir_qloss_h;...

        envir_UA_c*(tempout_c-tempin_c)/log((abs(tempout_c-...
        envir_temp)/abs(tempin_c-envir_temp)))-envir_qloss_c];

    elseif strcmp(obj.options{1},'option2')

        cer_res=obj.module.tcer_resist_group;

        %Define unknown array
        power1=x(1);
        q_h1=x(2);
        q_c1=x(3);
        te_temp_h=x(4);
        te_temp_c=x(5);
        tempout_h=x(6);
        tempin_c=x(7);
        deltaT_lm_h=x(8);
        deltaT_lm_c=x(9);
        envir_qloss_h=x(10);
        envir_qloss_c=x(11);
        q_ins1=x(12);
        te_temp_h_mod=x(13);
        te_temp_c_mod=x(14);


        %System of nonlinear equations
        F= [power1-q_h1+q_c1; ...

        mdotin_h*obj.specheat_h*(tempin_h-tempout_h)-...
        UA_h*deltaT_lm_h-envir_qloss_h;...

        mdotin_c*obj.specheat_c*(tempout_c-tempin_c)-...
        UA_c*deltaT_lm_c+envir_qloss_c;...

        q_ins1-(te_temp_h-te_temp_c)/insul_therm_res;...

        UA_h*deltaT_lm_h-q_h1-q_ins1;...

        UA_c*deltaT_lm_c-q_c1-q_ins1;...

        (tempin_h-tempout_h)/log((abs(tempin_h-te_temp_h)/...
        abs(tempout_h-te_temp_h)))-deltaT_lm_h;...

        (tempin_c-tempout_c)/log((abs(te_temp_c-tempout_c)/...
        abs(te_temp_c-tempin_c)))-deltaT_lm_c;...
```

```
        envir_UA_h*(tempin_h-tempout_h)/...
        log((abs(tempin_h-envir_temp)/abs(tempout_h-...
        envir_temp)))-envir_qloss_h;...

        envir_UA_c*(tempout_c-tempin_c)/...
        log((abs(tempout_c-envir_temp)/abs(tempin_c-...
        envir_temp)))-envir_qloss_c;...

        (te_temp_h-te_temp_h_mod)/cer_res-q_h1;...

        (te_temp_c_mod-te_temp_c)/cer_res-q_c1;...

        alpha*(alpha*(te_temp_h_mod-te_temp_c_mod)/...
        (load_res+elec_res))*te_temp_h_mod+1/therm_res*...
        (te_temp_h_mod-te_temp_c_mod)-0.5*elec_res*...
        (alpha*(te_temp_h_mod-te_temp_c_mod)/(load_res+...
        elec_res))^2-q_h1;...

        alpha*(alpha*(te_temp_h_mod-te_temp_c_mod)/...
        (load_res+elec_res))*te_temp_c_mod+1/therm_res*...
        (te_temp_h_mod-te_temp_c_mod)+0.5*elec_res*...
        (alpha*(te_temp_h_mod-te_temp_c_mod)/(load_res+...
        elec_res))^2-q_c1];
    end
end

function F = solve_zone_isothermal(obj,x,tempin_h,yy,mdotin_h)
    %This function solves a zone for an isothermal or constant heat
    %rate thermoelectric power unit and then solves successive
    %zones until the entire power unit is solved for. Inputs are
    %received from the other functions and this function is called
    %for by the compute function.

    %Define some local variables
    alpha = obj.module.talpha_group;
    therm_res = obj.module.ttherm_resist_group;
    elec_res = obj.module.telec_resist_group;
    load_res=obj.module.zone_load_res;
    insul_therm_res=obj.unit.insul_therm_res;
    UA_h=obj.unit.UA_h;
    envir_UA_h=obj.unit.envir_UA_h;
    envir_temp=obj.unit.envir_temp;

    if strcmp(obj.options{1},'option1')||...
            strcmp(obj.options{1},'option3')||...
            strcmp(obj.options{1},'option4')||...
            strcmp(obj.options{1},'option5')

        if strcmp(obj.options{2},'isothermal')
            te_temp_c=yy;
            q_c_tot1=x(4);
        elseif strcmp(obj.options{2},'constantheatrate')
            q_c_tot1=yy;
            te_temp_c=x(4);
        end
```

```
        %Define unknown array
        power1=x(1);
        q_h1=x(2);
        q_c1=x(3);
        te_temp_h=x(5);
        tempout_h=x(6);
        deltaT_lm_h=x(7);
        envir_qloss_h=x(8);
        q_ins1=x(9);

        %System of nonlinear equations
        F= [power1-q_h1+q_c1; ...

        alpha*(alpha*(te_temp_h-te_temp_c)/(load_res+elec_res))*...
        te_temp_h+1/therm_res*(te_temp_h-te_temp_c)-...
        0.5*elec_res*(alpha*(te_temp_h-te_temp_c)/...
        (load_res+elec_res))^2-q_h1;...

        alpha*(alpha*(te_temp_h-te_temp_c)/(load_res+elec_res))*...
        te_temp_c+1/therm_res*(te_temp_h-te_temp_c)+...
        0.5*elec_res*(alpha*(te_temp_h-te_temp_c)/...
        (load_res+elec_res))^2-q_c1;...

        mdotin_h*obj.specheat_h*(tempin_h-tempout_h)-...
        UA_h*deltaT_lm_h-envir_qloss_h;...

        q_ins1-(te_temp_h-te_temp_c)/insul_therm_res;...

        UA_h*deltaT_lm_h-q_h1-q_ins1;...

        (tempin_h-tempout_h)/log((abs(tempin_h-te_temp_h)/...
        abs(tempout_h-te_temp_h)))-deltaT_lm_h;...

        envir_UA_h*(tempin_h-tempout_h)/log((abs(tempin_h-...
        envir_temp)/abs(tempout_h-envir_temp)))-envir_qloss_h;...

        q_c1+q_ins1-q_c_tot1];

    elseif strcmp(obj.options{1},'option2')

        cer_res=obj.module.tcer_resist_group;

        if strcmp(obj.options{2},'isothermal')
            te_temp_c=yy;
            q_c_tot1=x(4);
        elseif strcmp(obj.options{2},'constantheatrate')
            q_c_tot1=yy;
            te_temp_c=x(4);
        end

        %Define unknown array
        power1=x(1);
        q_h1=x(2);
```

```
                q_c1=x(3);
                te_temp_h=x(5);
                tempout_h=x(6);
                deltaT_lm_h=x(7);
                envir_qloss_h=x(8);
                q_ins1=x(9);
                te_temp_h_mod=x(10);
                te_temp_c_mod=x(11);

                F= [power1-q_h1+q_c1; ...

                mdotin_h*obj.specheat_h*(tempin_h-tempout_h)-...
                UA_h*deltaT_lm_h-envir_qloss_h;...

                q_ins1-(te_temp_h-te_temp_c)/insul_therm_res;...

                UA_h*deltaT_lm_h-q_h1-q_ins1;...

                (tempin_h-tempout_h)/log((abs(tempin_h-te_temp_h)/...
                abs(tempout_h-te_temp_h)))-deltaT_lm_h;...

                envir_UA_h*(tempin_h-tempout_h)/log((abs(tempin_h-...
                envir_temp)/abs(tempout_h-envir_temp)))-envir_qloss_h;...

                q_c1+q_ins1-q_c_tot1;...

                (te_temp_h-te_temp_h_mod)/cer_res-q_h1;...

                (te_temp_c_mod-te_temp_c)/cer_res-q_c1;...

                alpha*(alpha*(te_temp_h_mod-te_temp_c_mod)/...
                (load_res+elec_res))*te_temp_h_mod+1/therm_res*...
                (te_temp_h_mod-te_temp_c_mod)-0.5*elec_res*...
                (alpha*(te_temp_h_mod-te_temp_c_mod)/(load_res+...
                elec_res))^2-q_h1;...

                alpha*(alpha*(te_temp_h_mod-te_temp_c_mod)/...
                (load_res+elec_res))*te_temp_c_mod+1/therm_res*...
                (te_temp_h_mod-te_temp_c_mod)+0.5*elec_res*...
                (alpha*(te_temp_h_mod-te_temp_c_mod)/(load_res+...
                elec_res))^2-q_c1];

            end
        end

    end

end
```

Appendix B

```
%Script for running tepowerunit.m model with data gathered from the
%experimental setup for the thermoelectric power unit
format long
clear classes
clear all
clc

equipment_lifetime = 30; %years
%[Ac_elec, Dc_elec, gas, oil, coal, thermal, flow, kinetic, potential]
costperkwh = [0.1,-0.1,.0,.18,.04,0,0,0,0];
costperc02 = [0,0,0];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Thermoelectric key parameters needed for option1
parameters.tepowerunit.module.ttherm_resist=1.5515;
parameters.tepowerunit.module.talpha=0.0460;
parameters.tepowerunit.module.width=0.04;
parameters.tepowerunit.module.thickness=0.0033;

%Hot side fins (rectangular straight fins)
parameters.tepowerunit.fins.t_h=0.001905;
parameters.tepowerunit.fins.l_h=.0254;
parameters.tepowerunit.fins.num_h=15;
parameters.tepowerunit.fins.k_h=209;
parameters.tepowerunit.fins.base_t_h=.012319;

%Cold side fins (rectangular straight fins)
parameters.tepowerunit.fins.t_c=0.001905;
parameters.tepowerunit.fins.l_c=.0254;
parameters.tepowerunit.fins.num_c=15;
parameters.tepowerunit.fins.k_c=209;
parameters.tepowerunit.fins.base_t_c=.012319+(3/8)*0.0254;

%Number of modules and number of zones
parameters.tepowerunit.unit.series=2;
parameters.tepowerunit.unit.parallel=2;
parameters.tepowerunit.unit.num=3;

%Insulation Between Base Plates
parameters.tepowerunit.unit.insul_k=0.06*4.425;
parameters.tepowerunit.unit.zone_to_mod_area_ratio=3;

%Thermal Contact Resistance [K*m^2/W]
parameters.tepowerunit.unit.therm_contact_res=0.0000544921231695565;

%Environmental Losses
parameters.tepowerunit.unit.uvalue=5;%17;
%1 inch of insulation around the fins
parameters.tepowerunit.unit.shell_t=.0254;
parameters.tepowerunit.unit.shell_k=0.05;%0.3;
parameters.tepowerunit.unit.envir_temp=300;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%Module 'option1','option3','option4','option5' Costs
parameters.tepowerunit.cost.specific.module=30; %$/module
%Fin Costs
parameters.tepowerunit.cost.specific.fin_material=4e5; %$/m3
parameters.tepowerunit.cost.fixed.fin_manufac_h=50; %$
parameters.tepowerunit.cost.fixed.fin_manufac_c=50; %$
%Insulation Costs
parameters.tepowerunit.cost.specific.insulation=350; %$/m3
%Cost Per Zone Area
parameters.tepowerunit.cost.specific.cost_per_zone_area=5; %$/zone area
%Other Costs
parameters.tepowerunit.cost.fixed.other=0; %$
%Fixed Assembly Costs
parameters.tepowerunit.cost.fixed.assembly=450; %$
parameters.tepowerunit.cost.costperkwh = costperkwh; %$/kwh
parameters.tepowerunit.cost.costperkwinst = 0;
parameters.tepowerunit.cost.costperc02 = costperc02;
parameters.tepowerunit.lifetime = equipment_lifetime; %yrs

parameters.tepowerunit.options={'option1','','straightfins_aligned','straight
fins_aligned'};

solver_inputs.fstr = '{tepowerunit(parameters.tepowerunit)}';

for i=1:4
    solver_inputs.n{i} =
fluid('N2,O2,Ar,CO2',[0.78,0.21,0.0077,0.0003],'GasMix','PT');
end
solver_inputs.cnmap = [4,1,2,3,4];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %235C/100CFM
all(1,1)=235;
all(1,2)=100;
% %Measured Power
all(1,3) = 13.493;
% mdot
all(1,4)=0.0553;
% Tc_in
all(1,5)=45.50;
% Tc_out
all(1,6)=67.48;
% Th_in
all(1,7)=247.02;
% Th_out
all(1,8)=218.97;
% Rload
all(1,9)=7.87;
%Re,mod
all(1,32)=2.422*1.2;

% %235C/75CFM
all(2,1)=235;
all(2,2)=75;
% %Measured Power
```

```
all(2,3) = 12.750;
% mdot
all(2,4)=0.0433;
% Tc_in
all(2,5)=39.03;
% Tc_out
all(2,6)=67.54;
% Th_in
all(2,7)=252.19;
% Th_out
all(2,8)=218.27;
% Rload
all(2,9)=7.81;
%Re,mod
all(2,32)=2.422*1.2;

% %235C/50CFM
all(3,1)=235;
all(3,2)=50;
% %Measured Power
all(3,3) = 9.055;
% mdot
all(3,4)=0.0291;
% Tc_in
all(3,5)=39.86;
% Tc_out
all(3,6)=77.30;
% Th_in
all(3,7)=247.85;
% Th_out
all(3,8)=205.86;
% Rload
all(3,9)=7.56;
%Re,mod
all(3,32)=2.422*1.2;

%200C/100CFM
all(4,1)=200;
all(4,2)=100;
%Measured Power
all(4,3) = 10.245;
% mdot
all(4,4)=0.0559;
% Tc_in
all(4,5)=46.16;
% Tc_out
all(4,6)=63.65;
% Th_in
all(4,7)=210.06;
% Th_out
all(4,8)=187.59;
% Rload
all(4,9)=7.88;
%Re,mod
all(4,32)=2.422*1.1;
```

```
% %200C/75CFM
all(5,1)=200;
all(5,2)=75;
% %Measured Power
all(5,3) = 9.709;
% mdot
all(5,4)=0.0431;
% Tc_in
all(5,5)=39.05;
% Tc_out
all(5,6)=62.13;
% Th_in
all(5,7)=212.79;
% Th_out
all(5,8)=185.03;
% Rload
all(5,9)=7.81;
%Re,mod
all(5,32)=2.422*1.1;

% %200C/50CFM
all(6,1)=200;
all(6,2)=50;
% %Measured Power
all(6,3) = 6.997;
% mdot
all(6,4)=0.0293;
% Tc_in
all(6,5)=40.06;
% Tc_out
all(6,6)=69.89;
% Th_in
all(6,7)=209.89;
% Th_out
all(6,8)=175.80;
% Rload
all(6,9)=7.52;
%Re,mod
all(6,32)=2.422*1.1;

% %150C/100CFM
all(7,1)=150;
all(7,2)=100;
% %Measured Power
all(7,3) = 5.357;
% mdot
all(7,4)=0.0566;
% Tc_in
all(7,5)=45.45;
% Tc_out
all(7,6)=57.23;
% Th_in
all(7,7)=157.49;
% Th_out
all(7,8)=142.21;
% Rload
```

```
all(7,9)=8.29;
%Re,mod
all(7,32)=2.422;

% %150C/75CFM
all(8,1)=150;
all(8,2)=75;
% %Measured Power
all(8,3) = 5.215;
% mdot
all(8,4)=0.0410;
% Tc_in
all(8,5)=37.71;
% Tc_out
all(8,6)=54.25;
% Th_in
all(8,7)=158.60;
% Th_out
all(8,8)=138.74;
% Rload
all(8,9)=7.77;
%Re,mod
all(8,32)=2.422;

% %150C/50CFM
all(9,1)=150;
all(9,2)=50;
% %Measured Power
all(9,3) = 3.820;
% mdot
all(9,4)=0.0288;
% Tc_in
all(9,5)=38.75;
% Tc_out
all(9,6)=59.22;
% Th_in
all(9,7)=155.81;
% Th_out
all(9,8)=131.99;
% Rload
all(9,9)=7.45;
%Re,mod
all(9,32)=2.422;

% %100C/100CFM
all(10,1)=100;
all(10,2)=100;
% %Measured Power
all(10,3) = 1.821;
% mdot
all(10,4)=0.0576;
% Tc_in
all(10,5)=44.88;
% Tc_out
all(10,6)=50.87;
% Th_in
```

```
all(10,7)=103.28;
% Th_out
all(10,8)=95.66;
% Rload
all(10,9)=8.15;
%Re,mod
all(10,32)=2.422;

% %100C/75CFM
all(11,1)=100;
all(11,2)=75;
% %Measured Power
all(11,3) = 1.865;
% mdot
all(11,4)=0.0419;
% Tc_in
all(11,5)=37.45;
% Tc_out
all(11,6)=46.22;
% Th_in
all(11,7)=103.63;
% Th_out
all(11,8)=93.24;
% Rload
all(11,9)=8.08;
%Re,mod
all(11,32)=2.422;

% %100C/50CFM
all(12,1)=100;
all(12,2)=50;
% %Measured Power
all(12,3) = 1.369;
% mdot
all(12,4)=0.0298;
% Tc_in
all(12,5)=37.97;
% Tc_out
all(12,6)=48.94;
% Th_in
all(12,7)=102.39;
% Th_out
all(12,8)=89.87;
% Rload
all(12,9)=8.15;
%Re,mod
all(12,32)=2.422;


for test=[1:12]
%1 is cfluidin
%2 is cfluidout
%3 is hfluidin
%4 is hfluidout

%Load Resistance per Zone
```

```
parameters.tepowerunit.module.zone_load_res=all(test,9); %ohms

%Module Key Parameter - Internal Electrical Resistance
parameters.tepowerunit.module.telec_resist=all(test,32); %ohms

solver_inputs.bcmap = [1,1,all(test,4);1,2,all(test,5)+273;1,3,101300;...
    3,1,0.94*all(test,4);3,2,all(test,7)+273;3,3,101300];
solver_inputs.xguess = [2,1,all(test,4);2,2,all(test,6)+273;2,3,101300;...
    4,1,0.94*all(test,4);4,2,all(test,8)+273;4,3,101300];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

solver_inputs.eps = 1e-6;
solver_inputs.h = 1e-7;
solver_inputs.minmax = 'max';
solver_inputs.removable = zeros(1,size(solver_inputs.cnmap,1));
dvlist = {'parameters.tepowerunit.fins.l_h'};

discrete = {[]};
for i =1:length(dvlist) %Get IC's for DVs from parameters structure
    dvguess(i) = eval(dvlist{i});
end

%Update relation for design variables
dvupdate = 'obj.parameters.tepowerunit.fins.l_h = obj.dvguess(1);';

%Formulate the cost function:
cost_function_def = {'0'; %A
                     '0'; %B
                     '[component_cost.power;component_cost.emissions]'; %C
                     '1'; %D
                     '0'; %E
                     'zeros(12,1)'; %F
                     [costperkwh,costperc02]; %Cost per unit for C
                     zeros(1,12); %Cost per unit for F
                     equipment_lifetime}; %Time by which to multiply B, C,
                     %E, and F

C = optimsolve(parameters,dvguess, solver_inputs,dvupdate,cost_function_def);
%Determine if the correct number of BCs and xguesses are supplied.

C.statecheck;

fmincon_options = optimset('UseParallel','always','Tolx', 1e-6);

%Set upper and lower constraints on each DV in the order that they appear
%in dvlist
lb=0.01;%lower bounds
ub= 0.1; %upper bounds

tstart=tic;
C.objective_f(dvguess);

all(test,10)=C.A.f{1,1}.power; %Model power
```

```
all(test,11)=(C.A.f{1,1}.power-all(test,3))/all(test,3)*100; %Percent diff
for measured and modeled power
all(test,12)=C.A.f{1,1}.q_h; %Modeled qh
all(test,13)=C.A.f{1,1}.q_c; %Modeled qc
all(test,14)=C.A.f{1,1}.q_ins; %Modeled qins
all(test,15)=C.A.f{1,1}.q_loss_h; %Modeled qloss,h
all(test,16)=C.A.f{1,1}.q_loss_c; %Modeled qloss,c
all(test,17)=C.A.f{1,1}.fins.h_resist; %Modeled Rh,fins
all(test,18)=C.A.f{1,1}.unit.UA_h; %Modeled UAh
all(test,19)=C.A.f{1,1}.fins.c_resist; %Modeled Rc,fins
all(test,20)=C.A.f{1,1}.unit.UA_c; %Modeled UAc
all(test,21)=C.A.nodes{1,1}.temp; %Final modeled Tc,in
all(test,22)=C.A.nodes{1,2}.temp; %Final modeled Tc,out
all(test,23)=C.A.nodes{1,3}.temp; %Final modeled Th,in
all(test,24)=C.A.nodes{1,4}.temp; %Final modeled Th,out
all(test,25)=1-(C.A.nodes{1,2}.temp-C.A.nodes{1,1}.temp)/(all(test,6)-
all(test,5)); %Cold side temp error
all(test,26)=1-(C.A.nodes{1,3}.temp-C.A.nodes{1,4}.temp)/(all(test,7)-
all(test,8)); %Hot side temp error
all(test,27)=C.A.f{1,1}.unit.insul_therm_res; %Thermal resistance of the
insulation between the hot and cold sides
all(test,28)=C.A.f{1,1}.unit.therm_contact_res; %Thermal contact resistance
(paste) K*m^2/W
all(test,29)=C.A.f{1,1}.unit.therm_contact_res/C.A.f{1,1}.unit.zone_mod_area;
%Thermal contact resistance (paste) K/W
all(test,30)=C.A.f{1,1}.unit.envir_UA_h; %Environmental UA for the hot side
all(test,31)=C.A.f{1,1}.unit.envir_UA_c; %Environmental UA for the cold side

end

%Header for Excel file
txt=[...
    'Test DegC                ';... %1
    'Test CFM                 ';... %2
    'Meas Power               ';... %3
    'Meas mdot                ';... %4
    'Meas Tc,in               ';... %5
    'Meas Tc,out              ';... %6
    'Meas Th,in               ';... %7
    'Meas Th,out              ';... %8
    'Meas Rload               ';... %9
    'Model Power              ';... %10
    'Model & Meas Power %Diff';... %11
    'Model qh                 ';... %12
    'Model qc                 ';... %13
    'Model qins               ';... %14
    'Model qloss,h            ';... %15
    'Model qloss,c            ';... %16
    'Model Rh,fins            ';... %17
    'Model UAh                ';... %18
    'Model Rc,fins            ';... %19
    'Model UAc                ';... %20
    'Tc,in Model Final        ';... %21
    'Tc,out Model Final       ';... %22
    'Th,in Model Final        ';... %23
    'Th,out Model Final       ';... %24
```

```
    'Cold Temps Error        ';... %25
    'Hot Temps Error         ';... %26
    'Model Rth,ins           ';... %27
    'Model Rth,con K*m^2/W   ';... %28
    'Model Rth,con K/W       ';... %29
    'Model UAh,envir         ';... %30
    'Model UAc,envir         ';... %31
    'Re,mod                  '];   %32

txt1=cellstr(txt);
txt2=txt1';

allcell=num2cell(all);
finalcell=[txt2;allcell];

%Write to an Excel file
xlswrite('Comparison.xlsx', finalcell, 'Test', 'A1:AF13');

%Elapsed time
telapsed=toc(tstart)
```